

Diplomarbeit
IFS 2006 - 0010

Modellierung der Dynamik eines Hexapods

angefertigt von

Savas Kurutas

September 2006



Kooperationspartner:



Prof. Dr.-Ing. Dieter Schramm
Dr.-Ing. Thorsten Brandt

Universität Duisburg-Essen
Campus Duisburg



Prof. Dr.-Ing. Karsten Lemmer
Dipl.-Ing. Martin Fischer

Deutsches Zentrum für Luft- und Raumfahrt
e. V.
Lilienthalplatz 7
380108 Braunschweig

Inhaltsverzeichnis

| | |
|--|-------------|
| Abkürzungsverzeichnis | VIII |
| Symbolverzeichnis | IX |
| Kurzfassung | XIII |
| Abstract | XIV |
| 1 Einleitung | 1 |
| 2 Theoretische Grundlagen | 4 |
| 2.1 Aufbau des Fahrsimulators | 5 |
| 2.1.1 Technische Daten | 6 |
| 2.1.2 Topologische Struktur | 7 |
| 2.1.3 Geometrische Beschreibung | 8 |
| 2.2 Kinematische Analyse | 10 |
| 2.2.1 Direkte Kinematik einer PKM | 11 |
| 2.2.2 Inverse Kinematik für PKM | 12 |
| 2.2.3 Direkte und inverse Kinematik für serielle Manipulatoren | 13 |
| 2.3 Arbeitsraumdefinition | 14 |
| 2.4 Begrenzungen des Arbeitsraums | 15 |
| 2.5 Geometrisch diskrete Arbeitsraumberechnung | 15 |
| 2.6 Analytisch geometrische Berechnung des Arbeitsraumrandes | 19 |
| 2.7 Analytische Methode zur Berechnung singulärer Stellungen bei PKM | 21 |
| 2.8 Berechnung von singulären Stellungen bei seriellen Manipulatoren | 23 |
| 2.9 Symmetrieeigenschaften bei der Arbeitsraumdarstellung | 24 |

| | | |
|----------|---|-----------|
| 3 | Simulationsergebnisse | 25 |
| 3.1 | 3-D Arbeitsraumdarstellung | 25 |
| 3.2 | 2-D Arbeitsraumdarstellung | 27 |
| 3.3 | Vergleich der graphischen Darstellung der beiden Untersuchungsmethoden | 29 |
| 3.4 | Abstandsberechnung zum Arbeitsraumrand | 30 |
| 3.4.1 | Berechnung für translatorische Abstände zum Arbeitsraumrand mit der diskreten Untersuchungsmethode | 31 |
| 3.4.2 | Berechnung der Schwenkwinkel zum Arbeitsraumrand mit der diskreten Untersuchungsmethode | 31 |
| 3.4.3 | Berechnung der horizontalen und vertikalen Abstände zum Arbeitsraumrand mit der geom. Untersuchungsmethode in der xy -Ebene | 32 |
| 3.4.4 | Simulationsergebnisse für die horizontale und vertikale Abstandsberechnung in der xy -Ebene | 34 |
| 3.4.5 | Berechnung der horizontalen und vertikalen Abstände zum Arbeitsraumrand mit der geom. Untersuchungsmethode in der yz -Ebene | 35 |
| 3.4.6 | Simulationsergebnisse für die horizontale und vertikale Abstandsberechnung in der yz -Ebene | 36 |
| 3.5 | Vergleich der beiden Untersuchungsmethoden | 36 |
| 3.5.1 | Auflistung der Berechnungszeit für die diskrete Untersuchungsmethode | 36 |
| 3.5.2 | Auflistung der Berechnungszeit der Schwenkwinkel mit der diskreten Untersuchungsmethode | 39 |
| 3.5.3 | Auflistung der Berechnungszeit für die geometrische Untersuchungsmethode | 41 |
| 3.5.4 | Bewertung für die Anwendung am Fahrsimulator | 43 |

| | | |
|----------|---|-----------|
| 4 | Funktionstest am DLR-Fahrsimulator | 44 |
| 4.1 | Auswertung der translatorischen Bewegungen der Plattform mit der diskreten Untersuchungsmethode | 44 |
| 4.1.1 | Funktionstest für translatorische Bewegungen der Plattform mit der geometrischen Untersuchungsmethode | 47 |
| 4.1.2 | Vergleich der diskreten mit der geometrischen Untersuchungsmethode | 48 |
| 4.1.3 | Funktionstest für Schwenkwinkel der Plattform | 49 |
| 4.2 | Vergleich des translatorischen Arbeitsraums | 51 |
| 4.3 | Fazit | 51 |
| 5 | Zusammenfassung und Ausblick | 53 |
| A | Algorithmus für den diskreten 3-D COW | 57 |
| B | Algorithmus für den geometrischen 3-D COW | 60 |
| C | Berechnung der Abstände in der xy-Ebene mit der geometrischen Untersuchungsmethode | 62 |
| D | Berechnung der Abstände des Schwenkwinkels φ im OW | 68 |

Abbildungsverzeichnis

| | | |
|------|---|----|
| 1.1 | Gough Plattform; Links: Foto; Rechts: Prinzipskizze | 1 |
| 2.1 | Fahrsimulator des DLR | 5 |
| 2.2 | Topologische Struktur | 8 |
| 2.3 | schematische Darstellung einer PKM | 10 |
| 2.4 | Mathematisch erchenbare Arbeitsräume | 12 |
| 2.5 | Serieller Roboter: Links: Kuka KR 16 CR; Rechts: zwei mögliche Gelenkkonfigurationen für eine vorgegebene Endeffektorposition | 13 |
| 2.6 | Ebener translatorischer Arbeitsraum für $z_i = z_h$; Oben: Draufsicht; Unten: Seitenansicht | 18 |
| 2.7 | Schnittmenge des ebenen Arbeitsraums bei $z_i = z_h$ | 18 |
| 2.8 | Verschiebung der Gelenkpunkte A_i und B_i | 19 |
| 2.9 | Schnittmenge der Kugelgleichungen für die yz -Ebene für $x = 0$ für ein System mit 3 Aktuatoren | 21 |
| 2.10 | Singuläre Stellungen im Arbeitsraum einer PKM | 22 |
| 2.11 | Mögliche singuläre Stellung einer PKM | 23 |
| 2.12 | Arbeitsraumrand für einen seriellen Manipulator; Links: Singuläre Stellungen; Rechts: technische Limitierung des Gelenkes | 23 |
| 2.13 | Symmetrieachsen einer Hexapodstruktur | 24 |
| 3.1 | translatorischer Arbeitsraum des Fahrsimulators ($\varphi = \theta = \psi = 0^\circ$); Links: Diskrete Unterteilung von 0.1 m; Rechts: Diskrete Unterteilung von 0.05 m | 25 |
| 3.2 | Arbeitsraum des Fahrsimulators; Links: Darstellung der eingrenzenden Kugelschalen; Rechts: Schnitt entlang der y -Achse | 26 |
| 3.3 | Ebener Arbeitsraum des Fahrsimulators bei einer Höhe von $z_{\text{ref}} = 3.95$ m. Links: $\varphi = 0^\circ$; Rechts $\varphi = 10^\circ$ | 27 |
| 3.4 | Ebener Arbeitsraum des Fahrsimulators bei einer Höhe von $z_h = 3.95$ m. Links: $\theta = 10^\circ$; Rechts $\psi = 10^\circ$ | 28 |

| | | |
|------|--|----|
| 3.5 | Ebener Arbeitsraum des Fahrsimulators bei einer Höhe von $z_i = 4.95$ m und $z_i = 2.95$ m | 28 |
| 3.6 | Vergleich der diskreten und geometrischen Untersuchungsmethode zur Arbeitsraumberechnung in der xz –Ebene; Links: diskrete Unterteilung 0.1 m; Rechts: diskrete Unterteilung 0.05 m | 29 |
| 3.7 | Vergleich der diskreten und geometrischen Untersuchungsmethode zur Arbeitsraumberechnung in der xy –Ebene; Links: diskrete Unterteilung 0.1 m; Rechts: diskrete Unterteilung 0.05 m | 30 |
| 3.8 | Arbeitsraum für eine konstante Höhe $z_i = z_{\text{ref}}$ mit vorgegebener konstanter Stellung $\varphi = \theta = \psi = 0^\circ$ der Plattform | 31 |
| 3.9 | Schwenken der Plattform um einen Referenzpunkt R | 32 |
| 3.10 | Geraden \mathbf{g}_v und \mathbf{g}_h ; Links: Gemeinsamer Schnittpunkt R in der xy –Ebene; Rechts: Abstandsberechnung in der xy –Ebene | 32 |
| 3.11 | Abstandsbestimmung in der xy –Ebene; Links: Schnittpunkte; Rechts: Bestimmung des kürzesten Intervalls | 35 |
| 3.12 | Abstandsbestimmung in der yz –Ebene; Links: Schnittpunkte; Rechts: Bestimmung des kürzesten Intervalls | 36 |
| 4.1 | Eingangssignale in $x - y - z$ –Richtung vom Fahrsimulator für translatorische Bewegungen der Plattform | 44 |
| 4.2 | Ausgangssignal vom Fahrsimulator für eine Diskretisierung von 0.5 m; Links: Betrag der Abstände in positiver und negativer x –Richtung; Rechts: Betrag der Abstände in positiver und negativer y –Richtung . | 45 |
| 4.3 | Ausgangssignal vom Fahrsimulator für eine Diskretisierung von 0.1 m; Links: Betrag der Abstände in positiver und negativer x –Richtung; Rechts: Betrag der Abstände in positiver und negativer y –Richtung . | 46 |
| 4.4 | Ausgangssignal vom Fahrsimulator für eine Diskretisierung von 0.05 m; Links: Betrag der Abstände in positiver und negativer x –Richtung; Rechts: Betrag der Abstände in positiver und negativer y –Richtung . | 46 |
| 4.5 | Ausgangssignal vom Fahrsimulator mit der geometrischen Untersuchungsmethode; Links: Betrag der Abstände in positiver und negativer x –Richtung; Rechts: Betrag der Abstände in positiver und negativer y –Richtung | 47 |

| | | |
|------|---|----|
| 4.6 | Überlagerung der Abstände der diskreten Untersuchungsmethode mit 0.05 m und der geometrischen Untersuchungsmethode; Links: gesamter Verlauf; Rechts: Ausschnitt bei 4 bis 10 s | 48 |
| 4.7 | Überlagerung der Abstände der diskreten Untersuchungsmethode mit 0.05 m und der geometrischen Untersuchungsmethode; Links: gesamter Verlauf; Rechts: Ausschnitt bei 4 bis 9 s | 48 |
| 4.8 | Eingangssignale vom Fahrsimulator für Neigung der Plattform; Links: Schwenkwinkel (φ, θ, ψ) ; Rechts: $x - y - z$ -Richtung | 49 |
| 4.9 | Ausgangssignal vom Fahrsimulator für Neigung φ der Plattform; Links: Ausgangssignal mit Berücksichtigung der translatorischen Bewegung; Rechts: Ausgangssignal aus konstanter Neutralposition | 50 |
| 4.10 | Translatorischer Arbeitsraum des Fahrsimulators ($\varphi = \theta = \psi = 0^\circ$); Links: Diskrete Unterteilung von 0.1 m; Rechts: Kontinuierlicher Arbeitsraum | 51 |

Tabellenverzeichnis

| | | |
|-----|---|----|
| 2.1 | Technische Daten des DLR-Fahrsimulators [18] | 6 |
| 3.1 | Diskrete Unterteilung für die Abstandsberechnung; Links: 0.1 m; Rechts: 0.05 m | 37 |
| 3.2 | Diskrete Unterteilung für die Abstandsberechnung; Links: 0.1 m; Rechts: 0.05 m | 38 |
| 3.3 | Diskrete Unterteilung für die Abstandsberechnung; Links: 0.1 m; Rechts: 0.05 m | 38 |
| 3.4 | Diskrete Unterteilung für den Schwenkwinkel φ ; Links: 0.5° ; Rechts: 0.1° | 40 |
| 3.5 | Diskrete Unterteilung für den Schwenkwinkel θ ; Links: 0.5° ; Rechts: 0.1° | 40 |
| 3.6 | Diskrete Unterteilung für den Schwenkwinkel ψ ; Links: 0.5° ; Rechts: 0.1° | 41 |
| 3.7 | Berechnungszeit nach der geometrischen Untersuchungsmethode für translatorische Abstände | 42 |
| 3.8 | Berechnungszeit nach der geometrischen Untersuchungsmethode für translatorische Abstände | 42 |

Abkürzungsverzeichnis

| | |
|-----|---|
| CAD | Computer Aided Design |
| COW | Constant Orientation Workspace |
| DLR | Deutsches Zentrum für Luft- und Raumfahrt |
| OW | Orientation Workspace |
| PKM | Parallel Kinematik Machine |
| R | Rotatorische Freiheitsgrade |
| P | Translatorische Freiheitsgrade |
| TCP | Tool Center Point |

Symbolverzeichnis

| | |
|------------------------------|---|
| A | Gelenkpunkt an der Basis |
| B_r | Gelenkpunkt an der Plattform |
| A' | Transformierter Gelenkpunkt |
| \mathbf{a} | Gelenkpunktvektor im Inertialsystem |
| \mathbf{a}'_i | Transformierter Gelenkpunktvektor |
| $\mathbf{a}_g, \mathbf{b}_g$ | Ortsvektoren zur Geradenkonstruktion |
| \mathbf{b}_r | Gelenkpunktvektor im körperfesten Koordinatensystem |
| C | körperfestes Koordinatensystem |
| f | Freiheitsgrade des Systems |
| f_G | Freiheitsgrade eines Gelenkes |
| \mathbf{g}_h | Gerade parallel zur y -Achse |
| \mathbf{g}_v | Gerade parallel zur x -Achse |
| i | Aktuatorindikator |
| \mathbf{I} | Einheitsmatrix |
| \mathbf{J} | Jacobi Matrix |
| \mathbf{l}_g | Gerade parallel zur z -Achse |

| | |
|----------------------------------|---|
| m_k | Multiplikator in der Geradendefinition |
| n_L | Anzahl der unabhängigen Schleifen |
| n_G | Anzahl der Gelenke |
| n_B | Anzahl der Körper |
| O | Inertialsystem |
| q | verallgemeinerte Koordinaten |
| q_0 | Ausgangslänge eines Aktuators |
| q_i | Kolbenausfahrlänge eines Aktuators |
| R | Schnittpunkt von zwei Geraden |
| R_i | Aktionsradius eines Aktuators |
| $\mathbf{R}(\mathbf{w})$ | Rotationsmatrix |
| $\mathbf{R}_{\varphi\theta\psi}$ | Durch Kardanwinkel beschriebene Rotationsmatrix |
| \mathbf{r} | Ortsvektor des Plattform-Mittelpunktes |
| \mathbf{u} | Differenzvektor |
| \mathbf{w} | Vektor der Weltkoordinaten |
| \bar{x} | Mittelwert |
| z_h | Fahrhöhe der Plattform |
| z_{ref} | Höhe der Neutralposition der Plattform |

| | |
|------------|---|
| β | Gelenkwinkel eines seriellen Manipulators |
| Δz | Abweichende Höhe von der Neutralpositionshöhe |
| σ | Standardabweichung |
| ψ | Orientierungswinkel um die x -Achse |
| θ | Orientierungswinkel um die y -Achse |
| φ | Orientierungswinkel um die z -Achse |
| ρ | Länge eines Aktuators |

Indizes bei Bewegungsgrößen

| | |
|-----|--|
| g | parallel zur z -Achse |
| h | parallel zur y -Achse |
| k | Allgemeines Indize für die Koordinaten einer Geraden |
| max | Maximalwert |
| min | Minimalwert |
| neg | Abstand in negativer Achsenrichtung |
| pos | Abstand in positiver Achsenrichtung |

| | |
|-----|---|
| r | Komponenten des Ortsvektors des Plattfrom-Mittelpunktes |
| ref | Referenzposition |
| v | parallel zur x -Achse |

Kurzfassung

In dieser Arbeit werden zwei Methoden vorgestellt, mit denen der Arbeitsraum eines Fahrsimulators berechnet werden kann. Die Berechnung basiert auf den kinematischen Eigenschaften des Mechanismus und erfolgt bei beiden Methoden über die inverse Kinematik. Eine diskrete Berechnung der Arbeitsraumzugehörigkeit aller Rasterpunkte innerhalb eines vorgegebenen Bereiches wird der analytischen Berechnung des Arbeitsraumrandes verglichen. Ausgehend von einer Referenzposition der Plattform soll der Abstand zum Arbeitsraumrand ausgegeben werden. Diese Informationen sollen für eine Vorpositionierung der Plattform während einer Simulationsfahrt als Basis dienen. Deshalb muss die Rechenzeit der Algorithmen möglichst klein sein, damit ausreichend Daten über den Abstand ausgegeben werden können. Insgesamt zeigt sich, dass die Genauigkeit und Rechenzeit der diskreten Untersuchungsmethode abhängig von der gewählten Schrittweite ist, während die analytische Untersuchungsmethode unabhängig davon genaue Ergebnisse liefert. Ein Vergleich bezüglich der Rechenzeit und Genauigkeit zeigt, dass die geometrische Untersuchungsmethode für die Berechnung von translatorischen Bewegungen der Plattform effektiver arbeitet. Für die rotatorischen Bewegungen der Plattform ist die Abstandsberechnung jedoch ausschließlich mit der diskreten Untersuchungsmethode möglich.

Schlagwörter:

- Arbeitsraumberechnung
- Inverse Kinematik
- Abstandsberechnung
- Diskrete Untersuchungsmethode
- Geometrische Untersuchungsmethode

Abstract

In this thesis two methods will be presented that can be used to calculate the workspace of a parallel kinematic machine. The calculation is based on the kinematic properties of a parallel manipulator. Both methods make use of the fundamentals of the inverse kinematics. An approach to calculate the workspace by the discretisation method will be compared with the calculation of the boundary of the workspace by the geometrical method. The distance between the platform and the boundary of the workspace should be displayed. This information shall be used for prepositioning the platform during a simulation drive. Thus the calculation time has to be very short in order to output enough information about the distance to the boundary of the workspace. It will be shown that the accuracy and the calculation time of the discret method depends on the stepsize. However the geometrical method gives always an exact information that is independent of the stepsize. A comparison between both methods based on the calculation time and accuracy shows that the geometrical method is more effective than the discrete method for the translational movement of the platform. The calculation of the distance to the boundary of the workspace for rotational movement of the platform is only possible using the discretization method.

Keywords:

- Workspace calculation
- Inverse kinematics
- Distance calculation
- Diskrete investigation method
- Geometrical investigation method

1 Einleitung

Die in vielen Fahrzeugen neuentwickelten Sicherheits- und Komfortsysteme genießen aufgrund ihrer technischen Komplexität und des daraus resultierenden größeren Funktionsumfangs immer höhere Aufmerksamkeit vom Fahrzeugkunden. Zur Entwicklung und Abstimmung dieser Systeme werden Testpersonen in Fahrsimulatoren eingesetzt. Der Vorteil eines Fahrsimulators ist, dass der Straßenverkehr realitätsgetreu nachgebildet werden kann, und Verkehrsszenarien wie zum Beispiel die Verkehrsdichte an einer Autobahnauffahrt vorgegeben und Fahrten beliebig oft wiederholt werden können. Durch gleichbleibende und reproduzierbare Versuchsbedingungen können genauere Ergebnisse erzielt werden. Die Testpersonen können Bewertungen über die Bedienungseigenschaften und Funktionsweise der getesteten Systeme geben, die bei der weiteren Entwicklung mitberücksichtigt werden können. Darüber hinaus kann u. a. das Fahrverhalten in kritischen Fahrsituationen untersucht werden, wobei die provozierten Unfallsituationen sowohl für den aktiven als auch für den passiven Verkehrsteilnehmer folgenlos bleiben.

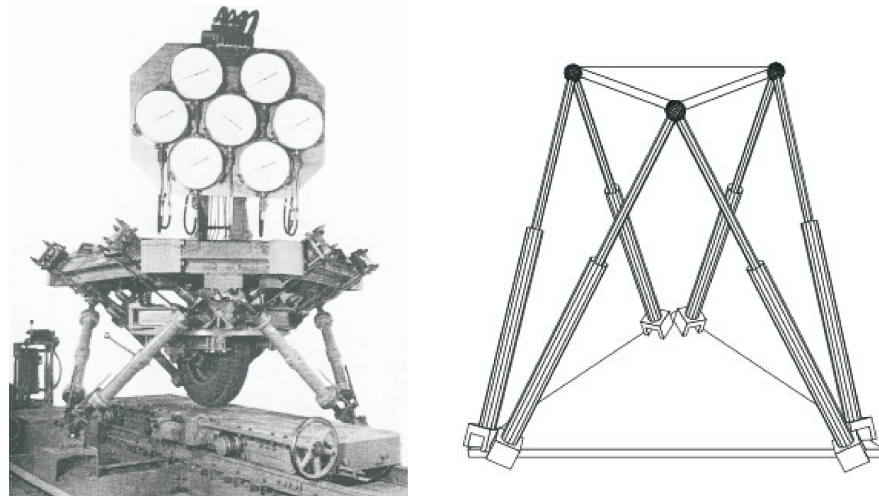


Bild 1.1: Gough Plattform; Links: Foto; Rechts: Prinzipskizze

Ein Fahrsimulator wie der am DLR-Institut für Verkehrsführung und Fahrzeugsteuerung genutzte, wird als *Parallel Kinematic Machine* (PKM) bezeichnet und hat die Struktur eines typischen Hexapods (vgl. Bild 1.1). Die erste PKM wurde 1956 von Gough [1] für die Reifenentwicklung entworfen. Stewart [2] hat 1965 auf dieser Basis einen Flugsimulator entwickelt, weil damit große Schwenkwinkel realisiert werden können.

Ein Vorteil der PKM gegenüber seriellen Robotern ist die hohe Struktursteifigkeit aufgrund der Verteilung der Kräfte auf mehrere Baugruppen. Da die Antriebe an der Basis angebracht sind, werden die zu bewegenden Massen insgesamt reduziert. Diese müssen nicht wie bei seriellen Robotern zusätzlich von den Aktuatoren getragen werden, wodurch sich höhere Beschleunigungen bei gleichzeitig weniger Energieverbrauch erreichen lassen. Zudem weisen die meisten PKMs, wie auch der DLR-Fahrsimulator, eine symmetrische Struktur auf, die aus gleichartigen Teilsystemen aufgebaut ist. Dadurch können baugleiche Komponenten für alle Antriebe verwendet werden [3].

Diesen Vorteilen stehen einige Nachteile wie die komplexe Bewegungsführung gegenüber. Zudem neigen PKMs dazu ungenau und nachgiebig zu sein [3]. Das ungünstige Verhältnis zwischen Maschinen- und Arbeitsraumvolumen wird durch die innovative Konstruktion (vgl. Bild 2.1) des DLR-Fahrsimulators, bei der die Kabine unterhalb der Plattform angebracht ist, reduziert.

Für eine effektivere Ansteuerung des Simulators ist die Kenntnis des zur Verfügung stehenden Arbeitsraums während einer Simulationsfahrt von Vorteil. Da der Arbeitsraum sich abhängig von der aktuellen Ausgangsposition und Orientierung dynamisch ändert, muss diese Information online immer wieder aktualisiert werden. Dadurch ergibt sich als Ziel, im Rahmen dieser Arbeit ein Modell für die dynamische Simulation des Arbeitsraums einer PKM zu erstellen. Es werden zunächst die theoretischen Grundlagen und ein geeigneter Ansatz zur Berechnung dieses Arbeitsraums in der Literatur recherchiert. Dieser Ansatz wird in einen Algorithmus umgesetzt und für die konkrete Anforderung der Aufgabenstellung weiterentwickelt. Das bedeutet, dass für eine beliebig angefahrene Position der Plattform die Abstände zum Arbeitsraumrand errechnet werden sollen. Diese Anforderung gilt zunächst für horizontale Abstände. Im nächsten Schritt werden die möglichen Neigungswinkel aus konstanter Position bis zum Arbeitsraumrand berechnet. Beide Lösungen sollen nach Möglichkeit in Echtzeit ermittelt werden. Das bedeutet, dass bei online Simulationsfahrten möglichst hinreichend viele Signale über den momentanen Abstand der Plattform zum Arbeitsraumrand ausgegeben werden soll.

In Kapitel 2 wird ein Überblick über die theoretischen Grundlagen zur Berechnung des Arbeitsraums von verschiedenen Autoren wiedergegeben. Dabei werden verschiedene Verfahren gegenübergestellt, und das am besten geeignet erscheinende ausgewählt und detailliert beschrieben.

Im 3. Kapitel werden die Simulationsergebnisse präsentiert. Darüber hinaus werden die Algorithmen hinsichtlich ihrer Rechenzeit verglichen.

In Kapitel 4 werden die Verfahren am DLR-Fahrsimulator angewandt. Dabei wird die

Funktionsweise der Algorithmen auf ihre Genauigkeit und Echtzeitfähigkeit getestet.

Das letzte Kapitel fasst die Ergebnisse dieser Arbeit zusammen und gibt einen Ausblick für weiterführende Arbeiten.

2 Theoretische Grundlagen

Für die Berechnung des Arbeitsraums einer PKM gibt es verschiedene Ansätze. In der Literatur wird dabei in der Regel von einer konstanten Orientierung der Plattform ausgegangen. Aus dieser vorgegebenen Position werden die Bewegungsmöglichkeiten in alle drei translatorischen Richtungen ermittelt. Mit diesen Ansätzen, die auf den geometrischen Eigenschaften einer PKM basieren, haben sich u. a. Fichter [15], Gossein [14] und Merlet [7] beschäftigt.

Eine andere Möglichkeit den Arbeitsraum einer PKM zu bestimmen, ist die Anwendung von CAD Tools. Bei diesen Programmen werden die geometrischen und technischen Daten der PKM eingegeben, wodurch der Aktionsradius der Aktuatoren berechnet und visualisiert werden kann. Es ist lediglich möglich die geometrischen Abmaße des Arbeitsraums darzustellen. Darüber hinaus ist die Berechnung des Arbeitsraums nicht in Echtzeit möglich.

Eine in dieser Arbeit angewandte Untersuchungsmethode basiert auf der Diskretisierung des Arbeitsraums. Hierbei wird ein 3-D Raum in eine vorgegebene Anzahl von Punkten unterteilt. In diesem Raster wird für jeden Punkt untersucht ob dieser zum Arbeitsraum gehört oder nicht. Die Genauigkeit bei diesem Verfahren hängt von der Wahl der Diskretisierung ab. Mit kleinen Schrittweiten können hohe Genauigkeiten erzielt werden, jedoch nur zu Lasten einer höheren Rechenzeit, die sich insbesondere in Hinblick auf die Echtzeitfähigkeit der Methode nachteilig auswirken kann. Jedoch spielt bei den ständig steigenden Rechnerleistungen diese Einschränkung eine immer kleinere Rolle. Davon abgesehen ist die Hauptanforderung in dieser Arbeit den Arbeitsraum in Echtzeit zu berechnen, und nicht eine möglichst hohe Genauigkeit zu erreichen. Somit kann eine relativ grobe Diskretisierung gewählt und damit eine ausreichende Schnelligkeit bei der Berechnung erzielt werden. Ein weiterer Nachteil bei diesem Verfahren ist, dass keine Aussage über Singularitäten zwischen den diskreten Punkten gemacht werden kann. Jedoch kann dieses Problem für die vorgegebene Anforderung aufgrund der sehr langen Rechenzeit nicht berücksichtigt werden.

Eine weitere Untersuchungsmethode den Arbeitsraum zu bestimmen ist der Ansatz über die geometrischen Eigenschaften. Dieser basiert zunächst wie die Diskretisierungsmethode auf der kinematischen Berechnung der zu untersuchenden PKM. Jedoch wird im Gegensatz zur diskreten Untersuchungsmethode nur die Begrenzung des Arbeitsraums über die Geometrie dargestellt. Ein möglicher Vorteil ist die kürzere Berechnungszeit und gleichzeitig, da es sich um eine analytische Untersuchungsmethode handelt, eine höhere Genauigkeit bei der Arbeitsraumberechnung.

Beide Verfahren zur Berechnung des Arbeitsraums sind analytisch. Beim diskreten Ansatz werden die kinematischen Beziehungen für diskrete Punkte analytisch berechnet und ausgewertet, während bei der geometrischen Untersuchungsmethode die Gleichungen geometrisch interpretiert werden können. Jedoch werden nicht die geometrischen Eigenschaften wie bei der Anwendung von CAD Tools zur Lösung herangezogen.

Der im Rahmen dieser Arbeit erarbeitete Ansatz basiert auf dem Konzept von Gosselin [14] und wird durch die Berechnung der inversen Kinematik gelöst. Dieser Ansatz ist von Merlet [9] für verschiedene Arten von PKMs weiter ausgearbeitet worden. Der auf geometrischen Eigenschaften basierende Ansatz zur Berechnung des Arbeitsraums wird als eines der effektivsten Verfahren bezeichnet [10].

Aufgrund dieser Kenntnisse wird über die kinematischen Eigenschaften der PKM eine diskrete und eine geometrische Untersuchungsmethode vorgestellt und auf den DLR-Fahrsimulator angewendet. Beide Verfahren werden anschließend in ihrer Genauigkeit und Rechenzeit miteinander verglichen. Ziel ist es die Berechnung in Echtzeit durchzuführen.

2.1 Aufbau des Fahrsimulators



Bild 2.1: Fahrsimulator des DLR

Der DLR-Fahrsimulator dient zu Testzwecken, bei denen der alltägliche Straßenverkehr simuliert werden soll. Mit dem Fahrsimulator sind große lineare Bewegungen von bis zu drei Metern möglich. Das Projektionssystem, mit einem Sichtbereich von $270^\circ \times 40^\circ$ und insgesamt 7500×1280 Pixeln erlaubt ein großes Blickfeld mit sehr hoher Auflösung. Weitere LCD-Displays sind auf der Rückbank und in den Aussenspiegeln integriert, damit die Testpersonen die rückwärtige Verkehrssituation beobachten kann. Um realitätsnahe Bedingungen zu schaffen, wurde ein komplettes Fahrzeug in die Kabine eingebaut. Da sich die Kabine unterhalb der Plattform befindet, ermöglicht es eine geringere Bauhöhe als bei anderen Fahrsimulatoren bei denen die Kabine oberhalb der Plattform angebracht ist. Dadurch wird ein einfacher Einstieg in das Fahrzeug ermöglicht. Das Soundsystem sorgt für ein realitätsnahes Motorgeräusch, welches sich kaum von dem eines realen Fahrzeuges unterscheidet. Diese Ausstattung ermöglicht den Testpersonen ein sehr realitätsnahes Fahrgefühl.

2.1.1 Technische Daten

| Fahrsimulator | | | |
|---------------|-------------------------|-------------------------|----------------------------|
| | Weg | Geschwindigkeit | Beschleunigung |
| Längs | ± 1.5 m | ± 2 m/s | ± 10 m/s ² |
| Quer | ± 1.4 m | ± 2 m/s | ± 10 m/s ² |
| Vertikal | $\pm 1,4$ m | ± 2 m/s | ± 10 m/s ² |
| Rollen | $-20^\circ / +21^\circ$ | $\pm 50^\circ/\text{s}$ | $\pm 250^\circ/\text{s}^2$ |
| Nicken | $\pm 21^\circ$ | $\pm 50^\circ/\text{s}$ | $\pm 250^\circ/\text{s}^2$ |
| Gieren | $\pm 21^\circ$ | $\pm 50^\circ/\text{s}$ | $\pm 250^\circ/\text{s}^2$ |

Tabelle 2.1: Technische Daten des DLR-Fahrsimulators

Die Bewegung der Plattform wird durch die Längenänderung der sechs hydraulisch angetriebenen Aktuatoren realisiert, welche durch passive Kardangelenke an der Basis und an der Plattform gelagert sind (vgl. Bild 2.1). Die minimale Aktuatorlänge bei eingefahrenen Kolben beträgt 4.45 m. Die Kolben erlauben einen Hub von bis zu zwei Meter, womit sich eine Maximallänge von 6.45 m ergibt. Die Simulationsfahrten starten von einer Neutralposition bei einer Höhe von $z_{\text{ref}} = 3.95$ m, bei der die Plattform die größten Bewegungsmöglichkeiten hat, da die Kolben hier genau halb ausgefahren sind und somit ein Hub von ± 1 m zur Verfügung steht. In Tabelle 2.1 sind die maximal darstellbaren Fahrwege, -geschwindigkeiten und -beschleunigungen des

DLR-Fahrsimulators aufgeführt. Der Simulator zählt mit seiner Nutzlast von ca. 1.3 Tonnen und den in Tabelle 2.1 aufgeführten Bewegungsmöglichkeiten zu den leistungsfähigsten Fahrsimulatoren Europas [18].

2.1.2 Topologische Struktur

In diesem Abschnitt wird das Verfahren zur Berechnung der Freiheitsgrade des Hexapods vorgestellt. Die Vorgehensweise ist sowohl auf serielle als auch auf parallele Manipulatoren anwendbar. Für die Berechnung der Freiheitsgrade wird die kinematische Struktur der PKM betrachtet. Der Unterschied zum physikalischen Modell liegt bei der Umgruppierung der Gelenkfreiheitsgrade des zylindrischen Gelenks am Aktuator und des Kardangelenks an der Plattform. Durch die Eigendrehung der Kolben um die Längsachse, welche sich mit der Achse des Kardangelenks im Gelenkmittelpunkt schneidet, wird dieses kinematisch als Kugelgelenk betrachtet, wobei das zylindrische Gelenk als Schubgelenk betrachtet wird (vgl. Bild 2.2). Zusätzlich wird ein virtuelles Gelenk eingeführt, das die räumlichen Bewegungsmöglichkeiten der Plattform repräsentiert, und dabei gerade den Weltkoordinaten

$$\mathbf{w} = \begin{bmatrix} x_r \\ y_r \\ z_r \\ \varphi \\ \theta \\ \psi \end{bmatrix}, \quad \in \mathbf{R}^6 \quad (2.1)$$

entspricht.

Die Anzahl der unabhängigen Schleifen

$$n_L = n_G - n_B = 19 - 13 = 6 \quad (2.2)$$

werden nach Hiller [17] berechnet, wobei n_G die Anzahl der Gelenke und n_B die Anzahl der Körper ist. Die Zahl der Freiheitsgrade für die Bewegung des 'tool center point' (TCP) berechnet sich aus der Summe der Freiheitsgrade aller Gelenke (vgl. Bild 2.2) nach der Formel

$$f = \sum_{i=1}^{n_G} f_{Gi} - 6n_L = (6 \times 1 + 6 \times 2 + 6 \times 3 + 1 \times 6) - 6 \times 6 = 6, \quad (2.3)$$

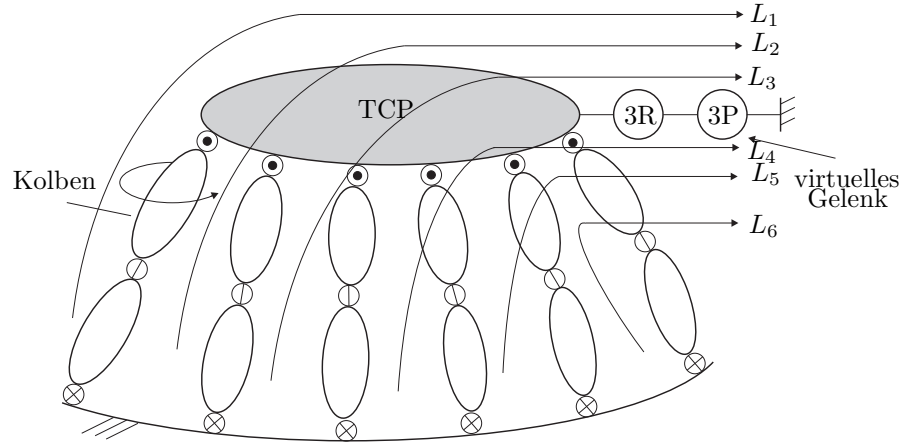


Bild 2.2: Topologische Struktur mit einer zulässigen Wahl für die unabhängigen Schleifen. Das Symbol \otimes bezeichnet dabei Kardangelenke, \odot steht für Schubgelenke und durch \odot sind Kugelgelenke gekennzeichnet. Das virtuelle Gelenk hat drei (R) rotatorische und drei (P) translatorische Freiheitsgrade

wobei f_{Gi} die Zahl der Freiheitsgrade der einzelnen Gelenke ist. Insgesamt besitzt das Hexapod demnach $f = 6$ Freiheitsgrade, womit sich die Maschine durch die verallgemeinerten Koordinaten $\mathbf{q} = [q_1, \dots, q_6]^T$ beschreiben lässt. Diese beschreiben die Ausfahrlänge der Kolben an den Schubgelenken, welche die einzig aktiven Stellglieder sind. Die verallgemeinerten Koordinaten \mathbf{q} stellen somit die Eingangsgrößen des Systems dar.

2.1.3 Geometrische Beschreibung

Die PKM besteht aus einer festen Basis und einer beweglichen Plattform, die über sechs Aktuatoren ($i = 1, \dots, 6$) miteinander verbunden sind. Die Position und Orientierung der Plattform wird über die Länge der Aktuatoren eingestellt. Die Struktur einer PKM ist schematisch in Bild 2.3 dargestellt.

Für die Berechnung des Arbeitsraums werden zwei Koordinatensysteme eingeführt. Der Ursprung des Inertialsystems O befindet sich am Mittelpunkt der Basis und das körperfeste Koordinatensystem C am Mittelpunkt der Plattform. Die Abstände der unteren Gelenkpunkte A_i zum Inertialsystem sind konstant und werden mit den Vektoren $\mathbf{a}_i = [a_{xi} \ a_{yi} \ a_{zi}]^T$ beschrieben. Die Abstände der oberen Gelenke zur Plattform-Mitte sind ebenfalls konstant und werden mit den Vektoren $\mathbf{b}_{r,i} = [b_{xi} \ b_{yi} \ b_{zi}]^T$ im

körperfesten Koordinatensystem beschrieben. Für die Beschreibung dieser Vektoren im Inertialsystem wird die Rotationsmatrix $\mathbf{R}_{\varphi\theta\psi}$ benötigt, welche die Verdrehung des körperfesten Koordinatensystems gegenüber dem Inertialsystem beschreibt:

$$\mathbf{R}_{\varphi\theta\psi} = \mathbf{R}_{\varphi}\mathbf{R}_{\theta}\mathbf{R}_{\psi}, \quad (2.4)$$

wobei die Kardanwinkel mit

$$\begin{aligned} \mathbf{R}_{\varphi} &= \begin{bmatrix} \cos(\varphi) & -\sin(\varphi) & 0 \\ \sin(\varphi) & \cos(\varphi) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{R}_{\theta} = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix}, \\ \mathbf{R}_{\psi} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\psi) & -\sin(\psi) \\ 0 & \sin(\psi) & \cos(\psi) \end{bmatrix} \end{aligned} \quad (2.5)$$

festgelegt sind [19]. Hierbei erfolgt die erste Rotation um die z -Achse (Gieren), gefolgt von der Rotation um die y -Achse (Nicken) und der Rotation um die x -Achse (Wanken).

Die Länge eines Aktuators $\boldsymbol{\rho}_i$ entspricht dem Abstand zwischen den Gelenkpunkten A_i und $B_{r,i}$, und kann durch die Gleichung

$$\boldsymbol{\rho}_i = \mathbf{r} - \mathbf{a}_i + \mathbf{R}_{\varphi\theta\psi}\mathbf{b}_{r,i} \quad (2.6)$$

berechnet werden, wobei $i = 1, \dots, 6$ für die jeweiligen Aktuatoren steht. Die Position des Plattform-Mittelpunktes gibt der Ortsvektor $\mathbf{r} = [x_r \ y_r \ z_r]^T$ an. Für eine übersichtlichere Schreibweise wird die Gleichung zu

$$\boldsymbol{\rho}_i = \mathbf{r} - \mathbf{u}_i \quad (2.7)$$

zusammengefasst, mit

$$\mathbf{u}_i = \mathbf{a}_i - \mathbf{R}_{\varphi\theta\psi}\mathbf{b}_{r,i}. \quad (2.8)$$

Die Komponenten des Vektors $\mathbf{u}_i = [u_{xi} \ v_{yi} \ w_{zi}]^T$ sind für eine feste Orientierung der Plattform konstant.

Der Betrag der Vektorgleichung 2.7 beschreibt die Länge des Aktuators

$$\rho_i = \|\boldsymbol{\rho}_i\| = \|\mathbf{r} - \mathbf{u}_i\|. \quad (2.9)$$

Für vorgegebene Werte der Position (x_r , y_r , z_r) und Orientierung (φ , θ , ψ) der Plattform stellt diese Gleichung auch die Lösung der inversen Kinematik dar (vgl. Kapitel 2.2.2).

Wird die Plattform ausschließlich translatorisch und parallel zur xy -Ebene bewegt ($\varphi = \theta = \psi = 0$), so vereinfacht sich die Rotationsmatrix $\mathbf{R}_{\varphi\theta\psi}$ zur Einheitsmatrix \mathbf{I} . Damit reduziert sich Gleichung (2.8) zu

$$\mathbf{u}_i = \mathbf{a}_i - \mathbf{b}_i, \quad (2.10)$$

da in diesem Fall $\mathbf{b}_{r,i} = \mathbf{b}_i$ ist.

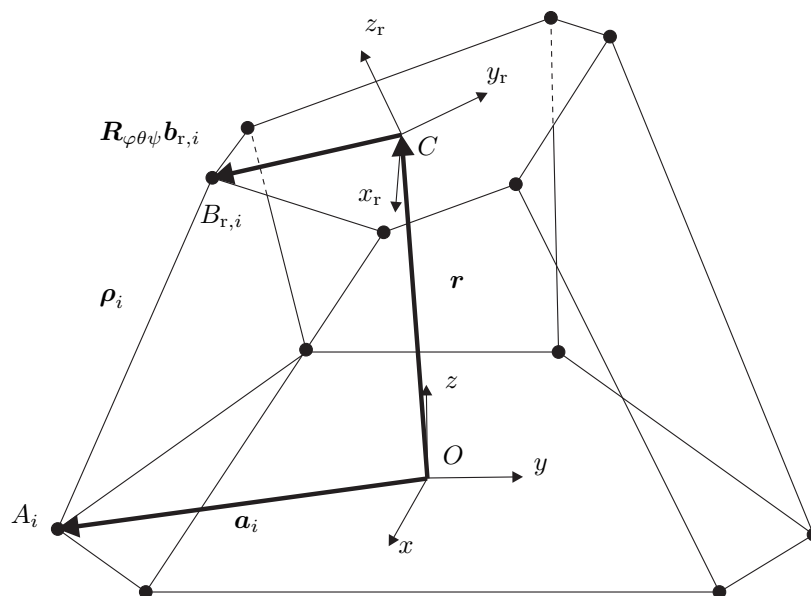


Bild 2.3: schematische Darstellung einer PKM

2.2 Kinematische Analyse

In diesem Abschnitt wird auf die kinematische Analyse der Arbeitsraumberechnung eingegangen, wobei im wesentlichen der Unterschied zwischen der direkten und in-

versen Kinematik näher erläutert wird. Dabei werden zunächst nur PKMs behandelt und anschließend der Unterschied zu seriellen Manipulatoren herausgearbeitet.

2.2.1 Direkte Kinematik einer PKM

Bei der direkten Kinematik wird aus vorgegebenen verallgemeinerten Koordinaten \mathbf{q} die Position und Orientierung \mathbf{w} der Plattform berechnet. Für serielle Roboter ist dies relativ einfach, aber für kinematische Schleifen ist die direkte Kinematik im Allgemeinen nicht geschlossen lösbar [5]. Im Gegensatz zu offenen kinematischen Ketten sind bei kinematischen Schleifen nicht mehr alle Eingangsgrößen voneinander unabhängig [17]. An der aus Kapitel 2.1.3 hergeleiteten kinematischen Gleichung 2.6 wird dies für den allgemeinen Fall

$$\boldsymbol{\rho}_i = \mathbf{r} + \mathbf{R}\mathbf{b}_{r,i} - \mathbf{a}_i, \quad (2.11)$$

näher erläutert. Da bei der direkten Kinematik die Aktuatorlänge die Eingangsgröße darstellt, wird zunächst der Betrag der Gleichung

$$\|\boldsymbol{\rho}_i\| = \sqrt{(\mathbf{r}(\mathbf{w}) + \mathbf{R}(\mathbf{w})\mathbf{b}_{r,i} - \mathbf{a}_i)^2}, \quad (2.12)$$

gebildet, um die Länge des Aktuators ρ_i zu erhalten. Die Aktuatorlänge $\rho_i = q_i + q_0$ hängt von der Ausgangslänge q_0 des Zylinders und von den verallgemeinerten Koordinaten q_i des Schubgelenks ab. Die Rotationsmatrix $\mathbf{R}(\mathbf{w})$ kann wie bereits erwähnt mithilfe von Kardan- bzw. Eulerwinkeln berechnet werden. Es gibt auch andere Methoden die Rotationsmatrix zu bestimmen, auf die aber nicht weiter eingegangen wird.

Für die direkte Kinematik müssen in dem nichtlinearen Gleichungssystem (2.12) alle sechs Koordinaten \mathbf{w} der Plattform bestimmt werden, wofür es keine geschlossene Lösung gibt. Daher lassen sich die Position und Orientierung nur über einen iterativen Lösungsansatz, wie z. B. mit dem Newton-Raphson Verfahren berechnen. Darüber hinaus ist zu beachten, dass bei gegebenem \mathbf{q} mehrere \mathbf{w} existieren, d. h. es gibt im Allgemeinen mehrere mathematisch mögliche Simulatorstellungen, die mit einer gegebenen Kombination verallgemeinerter Koordinaten übereinstimmen. Genauer gesagt sind insgesamt bis zu 40 Konfigurationen möglich [16]. Das liegt daran, dass es bei PKMs mehrere Zusammenbau Modi gibt, und die Position abhängig von

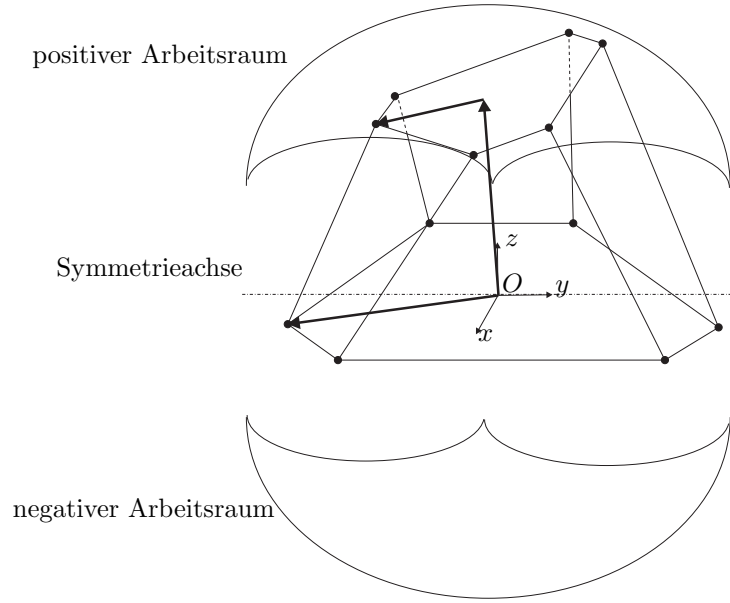


Bild 2.4: Mathematisch erchenbare Arbeitsräume

der Ausgangsposition ist [6]. Beispielhaft sei hierbei die gespiegelte Position der Plattform erwähnt, welche mathematisch betrachtet eine Lösung ergibt, jedoch physikalisch durch die räumliche Begrenzung bedingt nicht möglich ist (vgl. Bild 2.4).

Im weiteren wird nicht näher auf die direkte Kinematik eingegangen, da die Methode der inversen Kinematik besser für die Berechnung des Arbeitsraums geeignet ist.

2.2.2 Inverse Kinematik für PKM

Bei der inversen Kinematik werden für eine vorgegebene Position der Plattform \mathbf{w} die verallgemeinerten Koordinaten \mathbf{q} berechnet. Bei PKMs lässt sich die inverse Kinematik einfacher bestimmen als bei seriellen Manipulatoren, da von der Position und Orientierung der Plattformkoordinaten die Aktuatorlänge aufgrund der parallelen Struktur unabhängig vom Gesamtsystem berechnet werden kann [6]. Mit dem nicht-linearen Gleichungssystem (2.12) können explizit die verallgemeinerten Koordinaten \mathbf{q} für die vorgegebene Plattform Position \mathbf{w} berechnet werden

$$q_i = \pm \sqrt{(\mathbf{r}(\mathbf{w}) + \mathbf{R}(\mathbf{w})\mathbf{b}_{r,i} - \mathbf{a}_i)^2 - q_0}, \quad (2.13)$$

wobei die Gleichung allgemein zwei Lösungen hat. Jedoch ist aufgrund der technischen Limitierung der Aktuatoren nur das positive Ergebnis physikalisch sinnvoll.

2.2.3 Direkte und inverse Kinematik für serielle Manipulatoren

Bei seriellen Manipulatoren sind in der Regel die Aktuatoren und die Gelenke in einer offenen Kette aneinander gereiht. Der TCP wird durch diese Kette von Körpern und Gelenken bewegt, wobei diese im Gegensatz zu PKMs keine passiven Gelenke haben.

Für die Berechnung der direkten Kinematik existiert eine eindeutige Lösung für die TCP Stellung. Eine Möglichkeit ist dabei die Anwendung von Denavit und Hatenberg [20] welche eine analytische Beziehung zwischen verallgemeinerten Koordinaten und der Endeffektor Position über die Transformationsmatrix beschreibt.

Bei der inversen Kinematik gibt es dagegen keine eindeutige Lösung für die Berechnung der Gelenkkordinaten. Am Beispiel des zweiachsigen Roboters in Bild 2.5 ist relativ deutlich zu erkennen, dass für eine vorgegebene Endeffektorposition zwei Lösungen möglich sind. Bei einem sechsarmigen Knickarmroboter erhält man sogar bis zu acht Gelenkkonfigurationen für eine vorgegebene Position und Orientierung des Endeffektors.

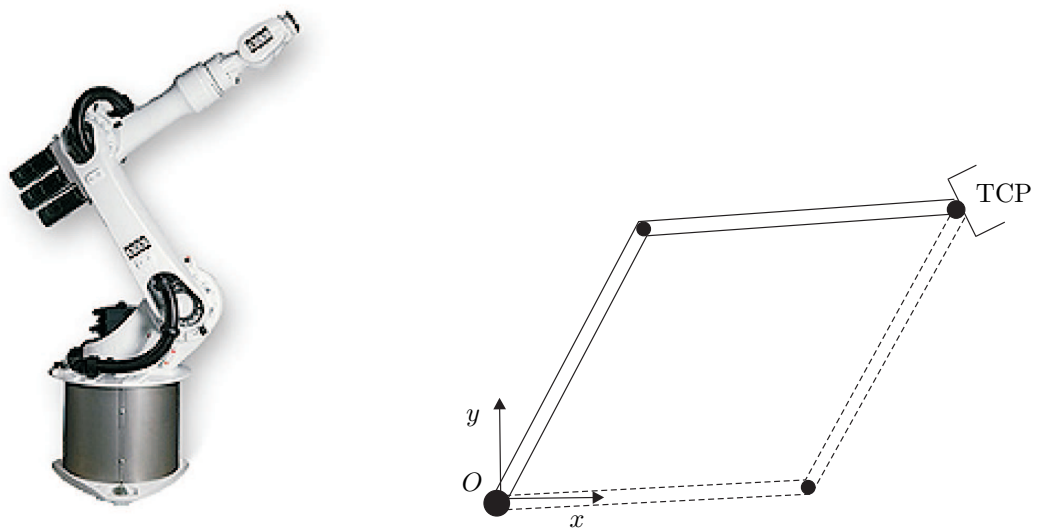


Bild 2.5: Serieller Roboter: Links: Kuka KR 16 CR; Rechts: zwei mögliche Gelenkkonfigurationen für eine vorgegebene Endeffektorposition

2.3 Arbeitsraumdefinition

Zunächst werden die Definitionen verschiedener Arbeitsrumtypen von PKMs vorgestellt und näher erläutert. Nach [8], [9] und [10] gilt:

Der *orientation workspace* (OW) ist als eine Menge definiert, bei der alle möglichen Rotationen für eine Referenzposition C des Plattform-Mittelpunktes möglich sind.

Der *constant orientation workspace* (COW) ist als eine Menge definiert, bei der alle translatorischen Positionen C mit einer konstanten Orientierung der Plattform erreichbar sind.

Der *total orientation workspace* (TOW) ist als ein Menge definiert, in dem alle möglichen Positionen eines Referenzpunktes bei Vorgabe eines Wertebereiches für φ θ ψ liegen.

Der *dextrous workspace* (DW) ist ein Spezialfall des (TOW), in dem für eine vorgegebene Position der Plattform alle möglichen Orientierungen bzw. Schwenkwinkel angegeben werden.

Der *maximal workspace* (MW) (auch *reachable workspace* genannt) ist die Menge aller durch mindestens eine Orientierung erreichbaren Positionen eines festgelegten Referenzpunktes.

Der *inclusive orientation workspace* (IOW) ist ein Spezialfall des MW, in dem ebenfalls die Menge aller durch mindestens eine Orientierung erreichbaren Positionen angegeben werden. Hierbei wird die Anzahl der möglichen Orientierungen für einen Bereich begrenzt. Für den ebenen Fall würde dies bedeuten, dass alle Winkel von 0 bis π betrachtet werden.

Es können noch weitere Arten von Arbeitsräumen definiert werden, um spezielle Bewegungsmöglichkeiten darzustellen.

Die in dieser Arbeit vorwiegend betrachteten Arbeitsräume sind der COW und der OW. Der Grund dafür liegt hauptsächlich darin, dass sich die Plattform des DLR-Fahrsimulators in der derzeitigen Software-Konfiguration bei Simulationsfahrten ausschließlich in einer Ebene z_{ref} bewegt. Von dieser Ausgangsposition wird während der Fahrt die Plattform translatorisch bewegt (COW). Darüber hinaus neigt sich die Plattform aus bestimmten Referenzpositionen um eine andauernde Beschleunigung und Geschwindigkeit auf die Testperson wirken zulassen. Deshalb ist es wichtig zu erfahren, in welcher Position die Plattform noch Reserven für maximal mögliche Schwenkwinkel hat (OW), und wie weit in der derzeitigen Position der Abstand zum

Arbeitsraumrand ist. Diese Daten können zur effektiveren Raumausnutzung dienen, die z. B. durch eine Vorpositionierung der Plattform erreicht wird.

2.4 Begrenzungen des Arbeitsraums

Der Arbeitsraum einer PKM kann allgemein durch mehrere Arten von Begrenzungen eingeschränkt werden, die den effektiv nutzbaren Bereich eingrenzen. Die wichtigsten sind im folgenden aufgelistet:

- Technische Limitierung durch minimale und maximale Aktuatorlängen
- Schwenkwinkelbegrenzung der Gelenke
- Kollision zwischen den Aktuatoren
- Singuläre Stellungen der Plattform

Der im Rahmen dieser Arbeit betrachtete Fahrsimulator des DLR ist konstruktiv so ausgelegt, dass keine Kollision zwischen den Aktuatoren möglich ist. Da die Berechnung in Echtzeit erfolgen soll, wird auf eine Überprüfung auf Singularitäten aufgrund der sehr langen Rechenzeit verzichtet. Weiterhin ist die PKM so konstruiert, dass die Aktuatorlängenlimitierung vor einer Schwenkwinkelbegrenzung der passiven Gelenke eintritt. Die maßgeblichen Größen bei der Begrenzung des Arbeitsraums des DLR-Fahrsimulators sind folglich die minimalen und maximalen Aktuatorlängen ρ_{\min} und ρ_{\max} . Deshalb wird zunächst nur die Prüfbedingung über die Begrenzung der minimalen und maximalen Aktuatorlängen abgefragt.

2.5 Geometrisch diskrete Arbeitsraumberechnung

Zur Berechnung der Aktuatorlänge wird die kinematische Gleichung 2.6 in skalare Form umgeschrieben, was geometrisch betrachtet einer Kugelgleichung entspricht

$$\rho_i^2 = (x_r - u_{xi})^2 + (y_r - v_{yi})^2 + (z_r - w_{zi})^2. \quad (2.14)$$

Für die diskrete Untersuchung wird ein 3-D Raum in eine vorgegebene Anzahl von diskreten Punkten unterteilt. Für jeden Punkt wird untersucht ob er zum Arbeitsraum gehört oder nicht. Dazu werden die Koordinaten jedes Punktes als Eingangsgrößen

in die Gleichung 2.14 eingesetzt. Es wird die benötigte Aktuatorlänge berechnet, um die vorgegebene Position ansteuern zu können.

Im nächsten Schritt gilt es zu untersuchen, ob die Aktuatoren zum Erreichen des Punktes die technische Limitierung der maximalen und minimalen Aktuatorlängen

$$\rho_{\min} \leq \rho_i \leq \rho_{\max}, \quad (2.15)$$

einhalten. Liegen alle sechs berechneten Aktuatorlängen für die vorgegebene Position innerhalb der zulässigen Werte wird der Punkt als Teil des Arbeitsraums gewertet. Sobald eine der Aktuatorlängen außerhalb der Bedingung liegt, wird der diskrete Punkt als nicht zulässig eingestuft. Diese Berechnung wird für alle Punkte des vorgegebenen Rasters durchgeführt, so dass die resultierende Punktwolke den Arbeitsraum darstellt.

Der Arbeitsraumrand ist erreicht, sobald mindestens einer der Aktuatoren die minimale oder maximale Länge annimmt und somit die Bedingung

$$\rho_i = \rho_{\min} \quad \text{oder} \quad \rho_i = \rho_{\max}, \quad (2.16)$$

erfüllt ist, wobei alle anderen Aktuatorlängen gleichzeitig innerhalb des erlaubten Bereiches liegen müssen.

Die Simulationsfahrten des DLR-Fahrsimulators finden bei einer Referenzhöhe $z_i = z_{\text{ref}} = 3.95$ m statt. Bei dieser Höhe sind die translatorischen Bewegungsmöglichkeiten der Plattform in der xy -Ebene am größten. Bisher sind noch keine Simulationsfahrten durchgeführt worden, bei denen es zu größeren Höhendifferenzen

$$\Delta z = z_{\text{ref}} - z_h \quad (2.17)$$

kommt, wie beispielsweise bei Simulationen für Steigungen oder Geländefahrten. Hierbei ist z_h die aktuell abweichende Fahrhöhe von der Neutralposition. Daraus abgeleitet ergibt sich die Anforderung den Arbeitsraum zunächst für eine ebene Arbeitsraumdarstellung $z_i = z_h = \text{konst.}$ darzustellen (vgl. Bild 2.6). Dazu werden die Gleichungen

$$(x_r - u_{xi})^2 + (y_r - v_{yi})^2 + (z_r - w_{zi})^2 = \rho_{\min,i}^2 \quad (2.18)$$

und

$$(x_r - u_{xi})^2 + (y_r - v_{yi})^2 + (z_r - w_{zi})^2 = \rho_{\max,i}^2 \quad (2.19)$$

umgestellt, wodurch sich diese zu Kreisgleichungen

$$(x_r - u_{xi})^2 + (y_r - v_{yi})^2 = R_{\min,i}^2 \quad (2.20)$$

und

$$(x_r - u_{xi})^2 + (y_r - v_{yi})^2 = R_{\max,i}^2, \quad (2.21)$$

mit

$$R_{\min,i} = \begin{cases} \rho_{\min}^2 - (z_h - w_{zi})^2, & \text{falls } \rho_{\min}^2 - (z_h - w_{zi})^2 > 0 \\ 0, & \text{sonst} \end{cases} \quad (2.22)$$

und

$$R_{\max,i} = \begin{cases} \rho_{\max}^2 - (z_h - w_{zi})^2, & \text{falls } \rho_{\max}^2 - (z_h - w_{zi})^2 > 0 \\ 0, & \text{sonst} \end{cases} \quad (2.23)$$

ergeben. Hierbei sind $R_{\min,i}$ die minimalen und $R_{\max,i}$ die maximalen Aktionsradien, die den Bereich des Aktuator i zwischen der minimal und maximal ausgefahrenen Position eingrenzen (vgl. Bild 2.6).

In Bild 2.6 ist ein Aktuator i dargestellt, der für eine vorgegebene Ebene z_h aufgrund der technischen Limitierung nur einen begrenzten Bereich anfahren kann. Es ist zu erkennen, dass bedingt durch die voll ausgefahrenen Länge des Aktuators i in negativer y -Richtung diese nur noch weiter nach unten verfahren kann (vgl. Bild 2.6, Punkte). Bei minimal eingefahrener Länge $\rho_{\min,i}$ müsste der Aktuator in positiver y -Richtung ebenfalls die Ebene verlassen. Der Bereich in dem der Aktuator i seinen Aktionsradius hat, wird somit durch $R_{\min,i}$ und $R_{\max,i}$ eingegrenzt.

In Bild 2.7 ist ein möglicher Bereich des Aktionsradius der Aktuatoren dargestellt. Die Kreise, die durch die Aktuatoren einen Bereich eingrenzen, ergeben zusammen die Schnittmenge die für die Plattform einen zulässigen Arbeitsraum der PKM parallel zur xy -Ebene darstellen. Die Begrenzung der Schnittmenge ist gleichzeitig der Arbeitsraumrand. Genauer gesagt ist die Schnittmenge eine Menge des Rasters, für die die Bedingung über die minimalen und maximalen Aktuatorlängen für alle Aktuatoren eingehalten werden.

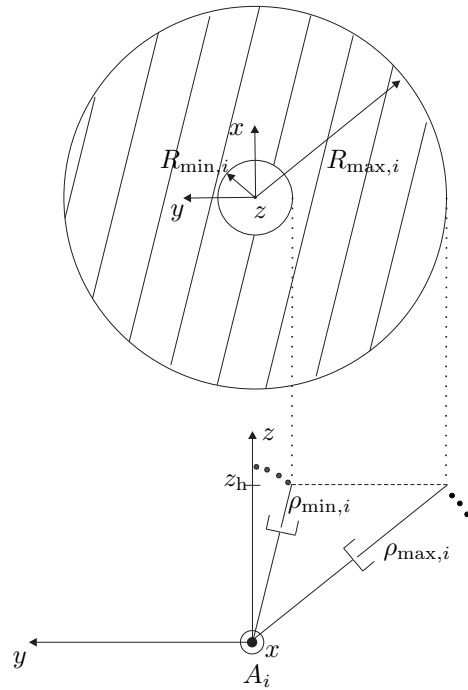


Bild 2.6: Ebener translatorischer Arbeitsraum für $z_i = z_h$; Oben: Draufsicht; Unten: Seitenansicht

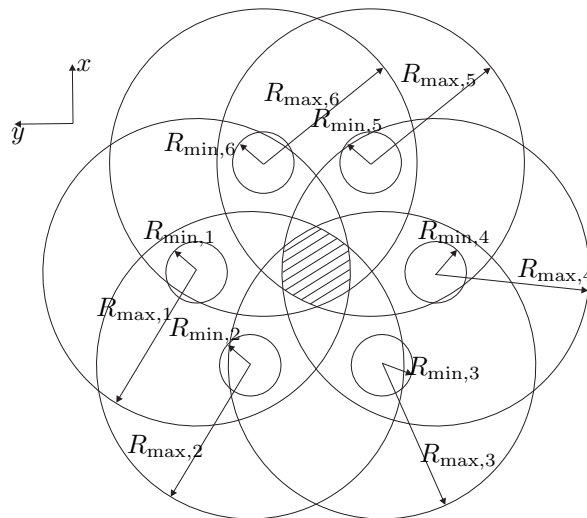


Bild 2.7: Schnittmenge des ebenen Arbeitsraums bei $z_i = z_h$

2.6 Analytisch geometrische Berechnung des Arbeitsraumes

Ein anderes geometrisches Verfahren von Merlet [8] zur Bestimmung des Arbeitsraums besteht in der Berechnung der Begrenzung des Arbeitsraums. Der wesentliche Unterschied liegt in der Untersuchungsmethode. So werden nicht diskrete Punkte des Arbeitsraums auf ihre Gültigkeit überprüft, sondern nur die exakten Arbeitsraumgrenzen anhand von Kugelgleichungen betrachtet. Der Vorteil hierbei gegenüber dem zuvor erläuterten Verfahren ist die wesentlich einfachere Darstellungsweise und vermutlich geringere Berechnungsdauer.

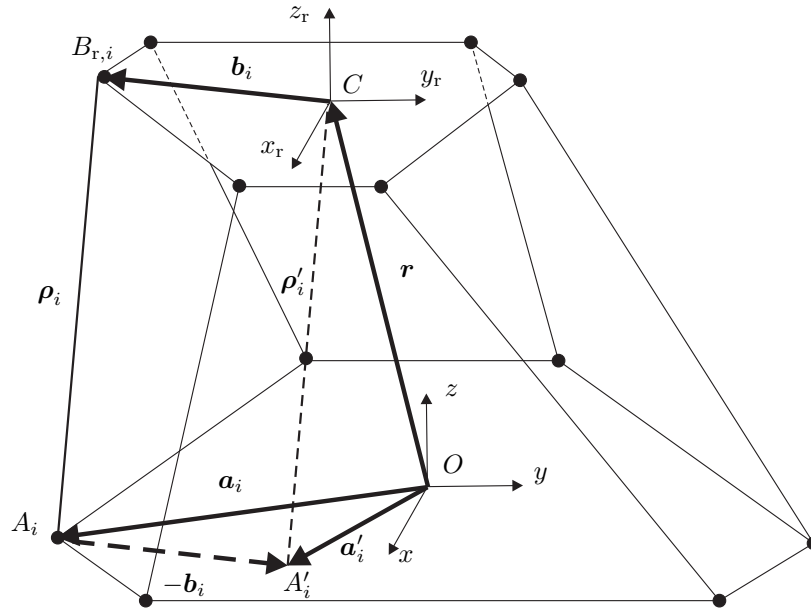


Bild 2.8: Verschiebung der Gelenkpunkte A_i und B_i

Die Gelenkpunkte der Basis werden entlang der Vektoren $-b_i$ (vgl. Bild 2.8) verschoben, so dass sich der obere jeweilige Gelenkpunkt im Plattform-Zentrum C befindet. Der neue untere Gelenkpunkt A'_i berechnet sich nun durch

$$a_i - b_i = a'_i, \quad (2.24)$$

wobei der neue Vektor ρ'_i parallel zu ρ_i verläuft. Durch die Verschiebung der Aktuatoren zum Plattform-Mittelpunkt wird die Geometrie der Plattform egalisiert. Um den

minimalen und maximalen Radius einer Kugel des Aktuators i zu berechnen, wird die quadratische Vektorgleichung

$$(\mathbf{r} - \mathbf{a}'_i)^2 = (\rho'_i)^2 \quad (2.25)$$

betrachtet. Unter Berücksichtigung, dass nur die Begrenzung des Arbeitsraums dargestellt werden soll, muss die Bedingung für die Radien der Aktuatoren

$$\rho_i = \rho_{\min} \quad \text{und} \quad \rho_i = \rho_{\max} \quad (2.26)$$

gelten, weil nur die minimalen und maximalen Verfahrswege der Kolben betrachtet werden. Diese Bedingungen werden für die skalare Form der Gleichung 2.25 verwendet, wodurch sich

$$(x_r - a'_x)^2 + (y_r - a'_y)^2 + (z_r - a'_z)^2 = \rho_{\max}^2 \quad (2.27)$$

und

$$(x_r - a'_x)^2 + (y_r - a'_y)^2 + (z_r - a'_z)^2 = \rho_{\min}^2 \quad (2.28)$$

ergeben.

Mit der Plattform können bei dieser Methode nur translatorische Bewegungen, die parallel zur xy -Ebene verlaufen, durchgeführt werden. Da bei einer Neigung der Plattform nicht alle transformierten Aktuatoren auf den Plattform-Mittelpunkt zeigen können, muss deshalb a'_z gleich Null sein. Durch den Bereich den alle 12 Kugelgleichungen gemeinsam eingrenzen, kann der Arbeitsraum bestimmt werden (vgl. Bild 2.9). Diese errechnet sich durch Auflösen der Gleichungen 2.27 und 2.28 nach

$$z_{\max,r} = \pm \sqrt{\rho_{\max}^2 - (x_r - a'_x)^2 - (y_r - a'_y)^2}, \quad (2.29)$$

und

$$z_{\min,r} = \pm \sqrt{\rho_{\min}^2 - (x_r - a'_x)^2 - (y_r - a'_y)^2}, \quad (2.30)$$

wobei immer die positiven Wurzeln betrachtet werden. Die negativen Wurzeln würden den Arbeitsraum unter der Boden-Ebene darstellen, was physikalisch nicht möglich ist.

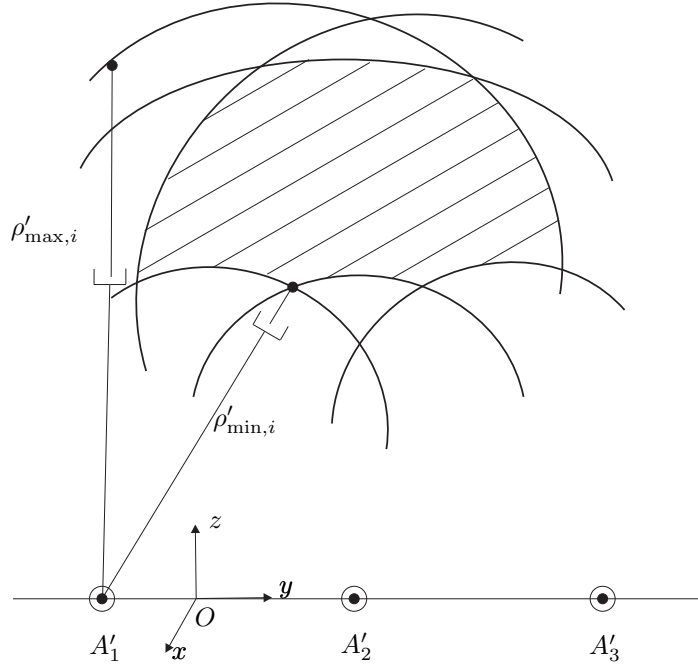


Bild 2.9: Schnittmenge der Kugelgleichungen für die yz -Ebene für $x = 0$ für ein System mit 3 Aktuatoren

2.7 Analytische Methode zur Berechnung singulärer Stellungen bei PKM

Ein anderer Weg zur Berechnung des Arbeitsraums, erarbeitet von Jo und Haug [13], geht von der Annahme aus, dass für einen Punkt am Arbeitsraumrand die Geschwindigkeit der Aktuatoren gleich Null wird. Hierbei zeigt der Geschwindigkeitsvektor normal zur Begrenzung des Arbeitsraums. Mit diesem Verfahren werden Singularitäten die sich im Arbeitsraum befinden entdeckt (vgl. Bild 2.10). Tritt eine kinematische Singularität ein, bei der sich die Gelenkachsen in einem Punkt schneiden, ist eine Bewegung der Plattform möglich, ohne dabei die Antriebe zu bewegen. In dieser Stellung versagt die PKM.

Solche Singularitäten können auch mit der Jacobi Matrix

$$\mathbf{J}_q = \frac{\partial \mathbf{q}}{\partial \mathbf{w}} \quad (2.31)$$

berechnet werden, weil bei dieser charakteristischen Stellung der Antriebe gilt

$$\mathbf{J}_q \partial \mathbf{w} = \partial \mathbf{q} = 0 \quad \text{für} \quad \partial \mathbf{w} \neq 0 \quad (2.32)$$

was äquivalent bedeutet, dass die Determinante der Jacobi Matrix gleich Null wird

$$\det \mathbf{J}_q = 0, \quad (2.33)$$

was wiederum gleichbedeutend ist mit einer Aktuatorgeschwindigkeit von Null. Ist der Arbeitsraumrand erreicht, liegt dies daran, dass einer der Aktuatoren seine Begrenzung $\rho_i = \rho_{\min}$ oder $\rho_i = \rho_{\max}$ erreicht hat. Diese Konfiguration kann jedoch von der Jacobi Matrix mathematisch nicht erfasst werden, weil nur eine technische Limitierung der PKM erreicht ist und bei verlängerten Aktuatoren diese Limitierung versetzt werden könnte. Es hängt also in diesem Fall ausschließlich von der konstruktiven Auslegung der PKM ab.

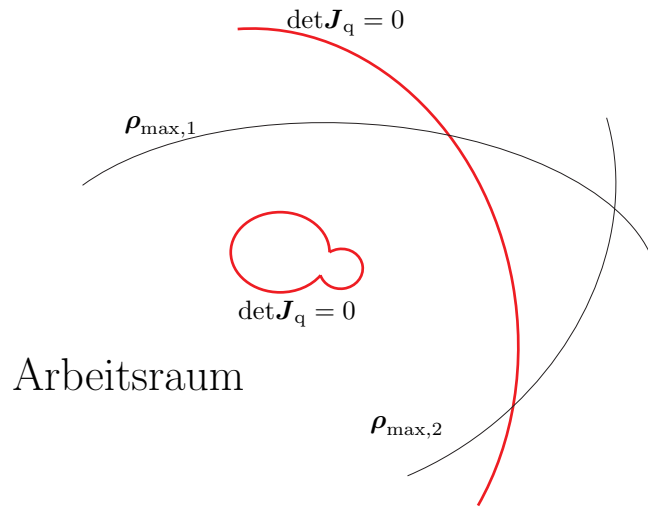


Bild 2.10: Singuläre Stellungen im Arbeitsraum einer PKM

Eine mögliche singuläre Stellung des DLR-Fahrsimulators wurde schon durch die konstruktive Auslegung unterbunden. In Bild 2.11 ist eine Position der Plattform dargestellt, in der sich die Aktuatorachsen schneiden, was eine singuläre Stellung ergeben würde. Diese kann jedoch vom DLR-Fahrsimulator nicht angefahren werden.

Einen Ansatz eine PKM so zu konstruieren, dass der Arbeitsraum frei von singulären Stellungen ist wird von Merlet [12] vorgestellt. Jedoch ist es bisher relativ schwierig

die Parameter zu bestimmen, weil dies noch einer der weniger erforschten Bereiche der Parallel Kinematik ist.

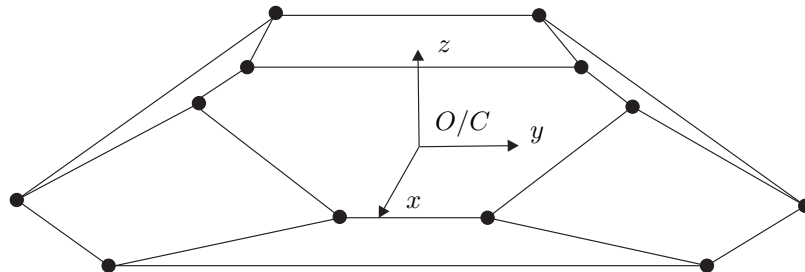


Bild 2.11: mögliche singuläre Stellung einer PKM

2.8 Berechnung von singulären Stellungen bei seriellen Manipulatoren

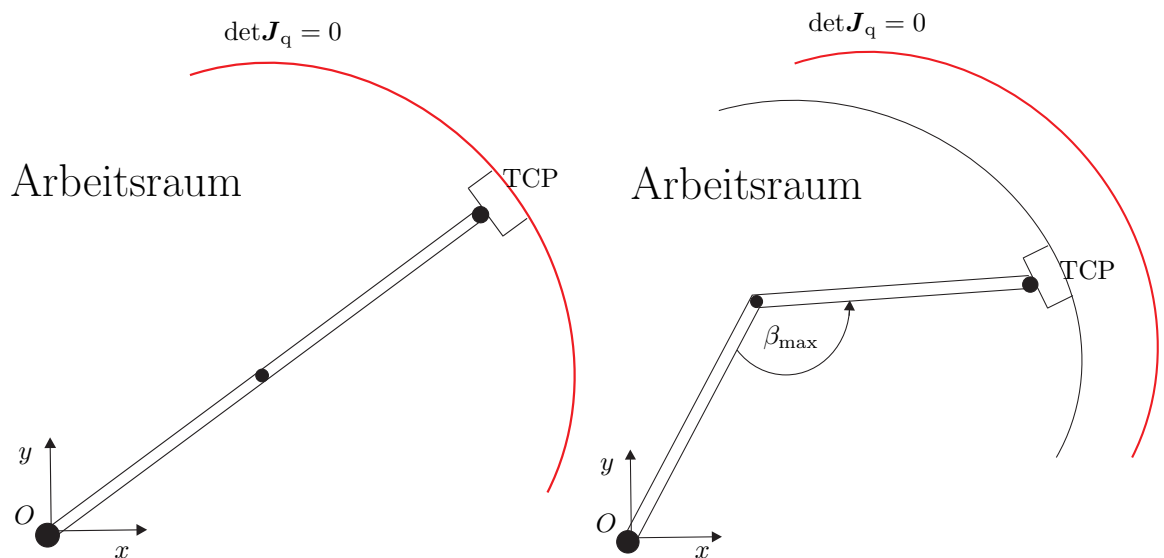


Bild 2.12: Arbeitsraumrand für einen seriellen Manipulator; Links: Singuläre Stellung; Rechts: technische Limitierung des Gelenkes

Bei seriellen Manipulatoren ist der Arbeitsraumrand bedingt durch die voll ausgestreckte Lage des Endeffektors auch gleichzeitig eine singuläre Stellung des Mecha-

nismus. Dieser Zustand kann, weil hierbei die Geschwindigkeit der Aktuatoren am Arbeitsraumrand gleich Null wird, von der Jacobi Matrix erfasst werden. Würde eine Winkelbegrenzung β_{\max} (vgl. Bild 2.12) der Aktuatoren die Bewegungsmöglichkeit des Manipulators einschränken, könnte eine singuläre Stellung verhindert werden. Denn dadurch sind die Aktuatoren nicht mehr senkrecht zu den Gelenkachsen orientiert. Jedoch wäre wiederum der Arbeitsraumrand nicht mehr über die Berechnung der Jacobi Matrix möglich.

Da der Schwerpunkt bei dieser Arbeit nicht auf der Untersuchung von Singularitäten liegt und dies auch den Rahmen der Arbeit überschreiten würde, wird auf eine weitere Bearbeitung dieses Verfahrens nicht näher eingegangen.

2.9 Symmetrieeigenschaften bei der Arbeitsraumdarstellung

Der DLR-Fahrsimulator hat insgesamt drei Symmetrieachsen (vgl. Bild 2.13). Aufgrund dieser symmetrischen Eigenschaften wäre es theoretisch ausreichend wenn der Arbeitsraum für einen Bereich von $0 - 60^\circ$ dargestellt werden würde. Um den gesamten Arbeitsraum zu erhalten müsste man nur diesen Abschnitt über die Symmetrieachsen entsprechend spiegeln. Jedoch wird im Rahmen dieser Arbeit darauf nicht näher eingegangen, weil es bezogen auf die echtzeitfähige Abstandsberechnung diese Fallunterscheidung vermutlich zu keiner wesentlich kürzeren Berechnungszeit führen würde. Sobald ein Referenzpunkt außerhalb des modellierten Arbeitsraumausschnittes liegen würde, müsste dieser erst durch die Spiegelungen erfasst werden. Dies würde zu zeitaufwändigen Fallunterscheidungen im Algorithmus führen, womit vermutlich der Vorteil der Teilmodellierung kompensiert werden würde.

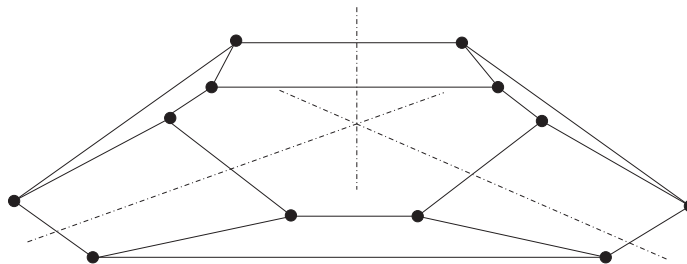


Bild 2.13: Symmetrieachsen einer Hexapodstruktur

3 Simulationsergebnisse

In diesem Kapitel werden die Simulationsergebnisse des erstellten Modells präsentiert. Dabei werden u. a. für verschiedene konstante Positionen und Orientierungen der Plattform die Bewegungsmöglichkeiten graphisch dargestellt. Es muss berücksichtigt werden, dass der Arbeitsraum des DLR-Fahrsimulators sechs Freiheitsgrade hat, so dass der Arbeitsraum in einem 6-dimensionalen Raum dargestellt werden müsste. Dies ist jedoch nicht anschaulich darstellbar. Deshalb müssen jeweils drei Koordinaten als konstant angenommen werden, um den Arbeitsraum graphisch im 3-D Raum veranschaulichen zu können.

3.1 3-D Arbeitsraumdarstellung

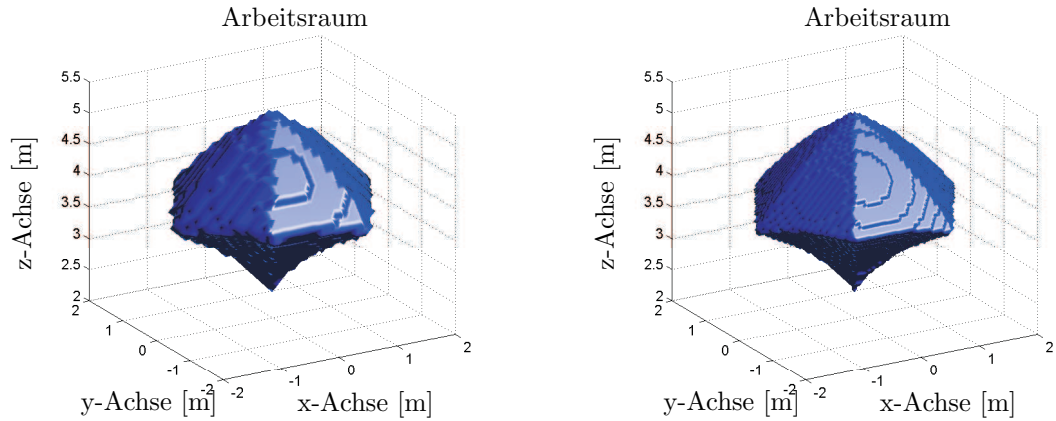


Bild 3.1: translatorischer Arbeitsraum des Fahrsimulators ($\varphi = \theta = \psi = 0^\circ$); Links: Diskrete Unterteilung von 0.1 m; Rechts: Diskrete Unterteilung von 0.05 m

In Bild 3.1 ist der Arbeitsraum des DLR-Fahrsimulators für eine konstante horizontale Orientierung (COW) der Plattform dargestellt. Das bedeutet, dass die Winkel sich nicht ändern. Zu sehen ist ein Vergleich des Arbeitsraums für verschiedene Diskretisierungsschrittweiten. Die schirmähnliche Form wird mit steigender Diskretisierungszahl immer deutlicher. Es ist zu erkennen, dass für die Ebene $z_i = z_{\text{ref}} = 3.95$ m, die translatorischen Bewegungsmöglichkeiten am größten sind. Diese Ebene enthält auch als Neutralposition und wird bei Beginn der Versuchsfahrten des DLR-Fahrsimulators angefahren. Bei einer senkrechten Abweichung von der Neutralposition nehmen die horizontalen Bewegungsmöglichkeiten stetig ab. Das hängt von der aktuellen Position

der Kolben und dem restlichen zur Verfügung stehenden Hub ab.

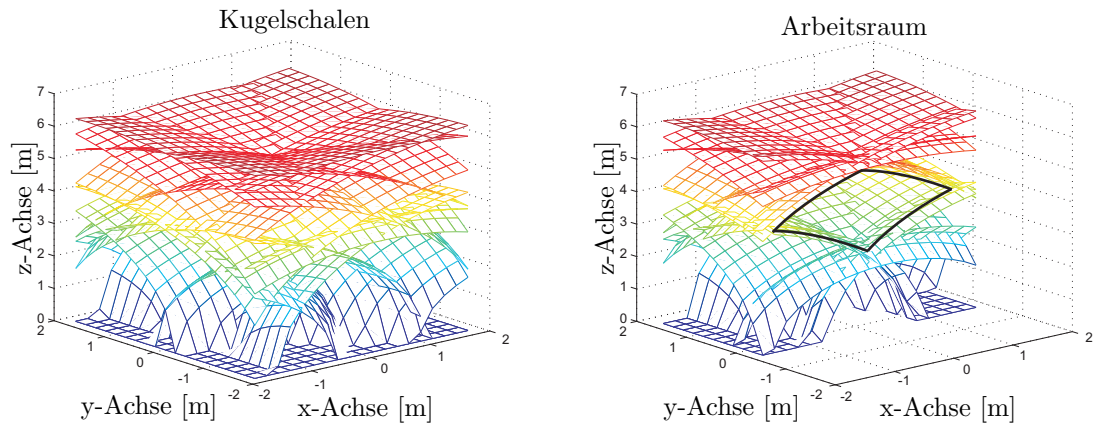


Bild 3.2: Arbeitsraum des Fahrsimulators; Links: Darstellung der eingrenzenden Kugelschalen; Rechts: Schnitt entlang der y -Achse

Das in Bild 3.2 eingegrenzte Volumen stellt ebenfalls einen COW dar. Dieser ist nach der analytischen Untersuchungsmethode graphisch dargestellt. Die in Kapitel 2.6 hergeleiteten Kugelgleichungen spannen Oberflächen auf, deren Gesamtschnittmenge die Bewegungsmöglichkeiten des Plattform-Mittelpunktes eingrenzen. Ein Schnitt durch die y -Achse veranschaulicht, dass sich auch hier eine schirmähnliche Form ergibt. Es wird nur die Begrenzung des Arbeitsraums dargestellt. Das eingegrenzte Volumen stellt für alle Aktuatoren gleichzeitig erreichbare Position der Plattform für eine konstante horizontale Orientierung dar. Diese stellt für gleiche geometrischen Daten wie bei der diskreten Untersuchungsmethode berechneten Arbeitsraum ein exaktes Ergebnis dar.

Für komplexere Versuchsfahrten, wie z. B. eine simulierte Geländefahrt bei der mehrere Ebenen angefahren werden müssen, ist die 3-D Darstellung des Arbeitsraums eine wertvolle Information. Diese Daten können für eine veränderte Vorpositionierung genutzt werden, falls der bisher zur Verfügung stehende Arbeitsraum für eine weitere Steigung aus der momentanen Position nicht mehr ausreicht. Darüber hinaus kann bestimmt werden, wie hoch die Plattform noch bewegt werden kann. An diesem Beispiel wird nochmal deutlich wie wichtig die echtzeitfähige Berechnung des Arbeitsraums ist, um den Arbeitsraum effektiv nutzen zu können.

3.2 2-D Arbeitsraumdarstellung

Die Versuchsfahrten starten wie bereits erwähnt in der xy -Ebene mit $z_i = z_{\text{ref}}$ von der aus die Plattform translatorisch und rotatorisch die größten Bewegungsmöglichkeiten hat. Um den Einfluss von veränderten Orientierungen auf die Bewegungsmöglichkeiten des Fahrsimulators zu untersuchen, werden 2-D Arbeitsräume für verschiedene Neigungen der Plattform dargestellt (vgl. Bild 3.3). Weiterhin werden Arbeitsräume für verschiedene Höhen z_i miteinander verglichen. Darüber hinaus stellt die 2-D Arbeitsraumberechnung die Basis für die Abstandsberechnung zum Arbeitsraumrand während einer Versuchsfahrt dar.

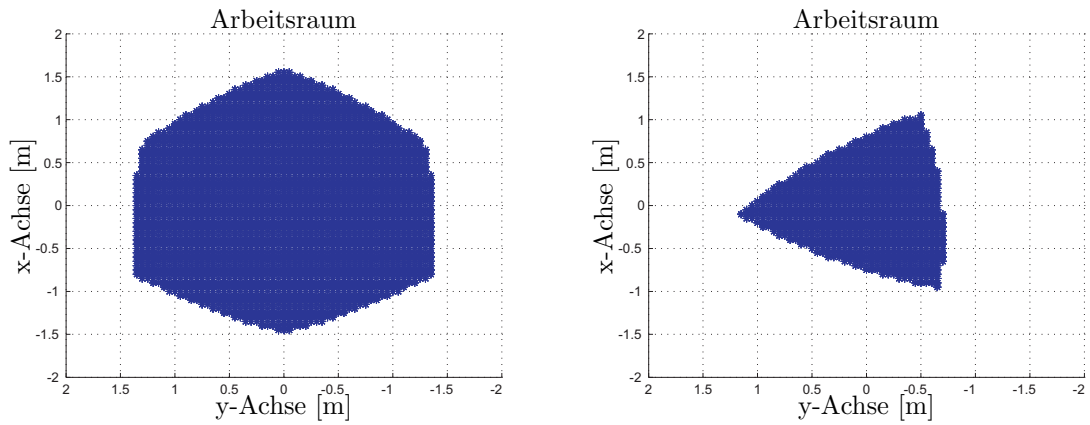


Bild 3.3: Ebener Arbeitsraum des Fahrsimulators bei einer Höhe von $z_{\text{ref}} = 3.95$ m.
Links: $\varphi = 0^\circ$; Rechts $\varphi = 10^\circ$

In Bild 3.3 ist ein ebener Bereich des Arbeitsraums für eine Höhe von $z_{\text{ref}} = 3.95$ m dargestellt. Für eine Orientierung der Plattform um $\varphi = 10^\circ$ ist eine Änderung des Arbeitsraums zu erkennen. Das lässt sich durch die veränderte Stellung der Aktuatoren erklären, die durch die Neigung bereits in der Ausgangslage ($x = y = 0, z_i = z_{\text{ref}}$) aus ihrer Neutralposition $q_i = 1$ m verfahren sind. Dies hat zur Folge, dass die minimale und maximale Aktuatorlänge bei einem oder mehreren Aktuatoren bereits bei kleineren x - bzw. y -Positionen erreicht wird.

In Bild 3.4 sind die Verformungen des Arbeitsraums für die Winkel θ und ψ bei einer Änderung von je 10° dargestellt. Durch die Drehung der Plattform sind die unterschiedlichen Bereiche des Arbeitsraums zu erkennen. Bei der Verdrehung um

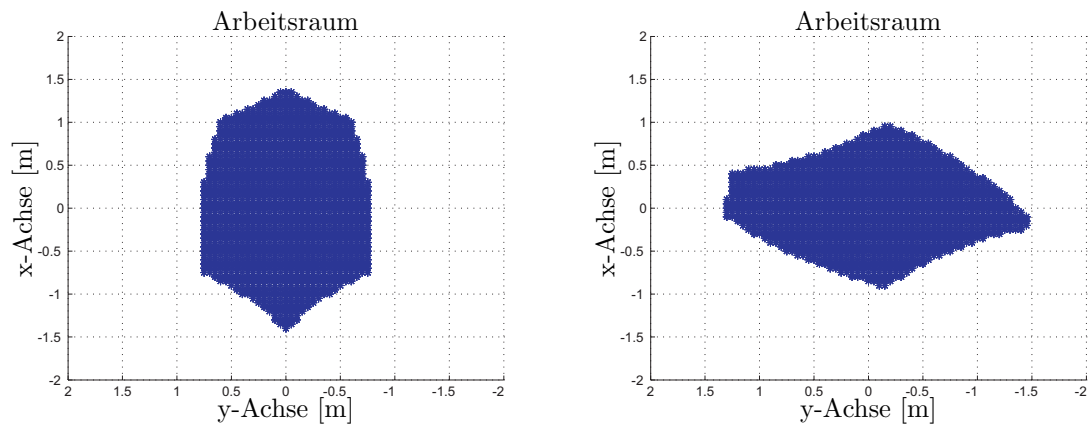


Bild 3.4: Ebener Arbeitsraum des Fahrsimulators bei einer Höhe von $z_h = 3.95$ m.
Links: $\theta = 10^\circ$; Rechts $\psi = 10^\circ$

die y -Achse (θ) stehen mehr Bewegungsmöglichkeiten entlang der x -Achse zur Verfügung. Dieser Zusammenhang ist auch für die Verdrehung um die x -Achse (ψ) zu erkennen, wobei hier die Fläche des Arbeitsraums sich entlang der y -Achse ausbreitet. Diese Verformung ist auf die Verstellung der Kolben von den Aktuatoren zurückzuführen.

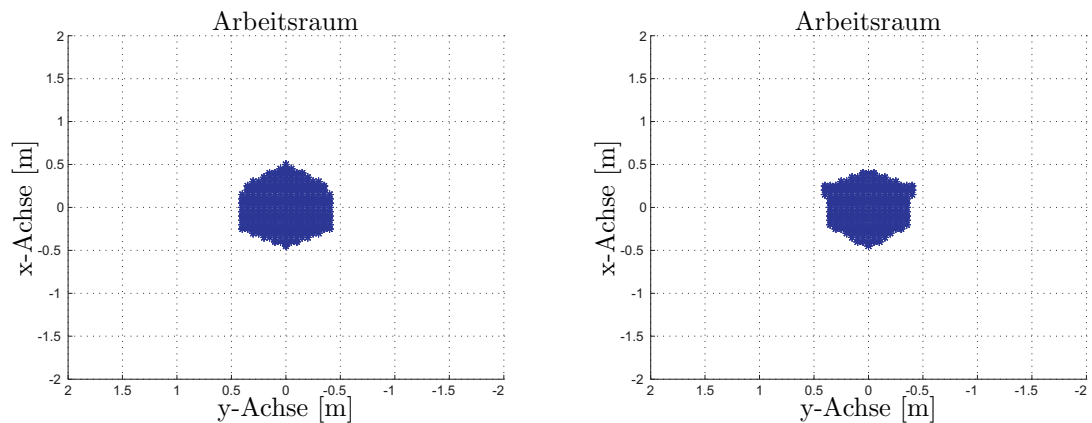


Bild 3.5: Ebener Arbeitsraum des Fahrsimulators für verschiedene Höhen. Links $z_i = 4.95$ m; Rechts $z_i = 2.95$ m

In Bild 3.5 sind die ebenen Arbeitsräume für verschiedene Höhen dargestellt. Die

Fläche oberhalb und unterhalb von der Neutralposition z_{ref} nimmt sichtbar ab. Das hängt wiederum von dem jeweils zur Verfügung stehenden Hub der Aktuatoren ab, die abweichend von der Neutralposition früher einsetzen. Da von dieser Position der zur Verfügung stehende Hub der Aktuatoren insgesamt eingeschränkt wird, ist deutlich warum bei verschiedenen Höhen von z_i die Bewegungsmöglichkeiten abnehmen.

3.3 Vergleich der graphischen Darstellung der beiden Untersuchungsmethoden

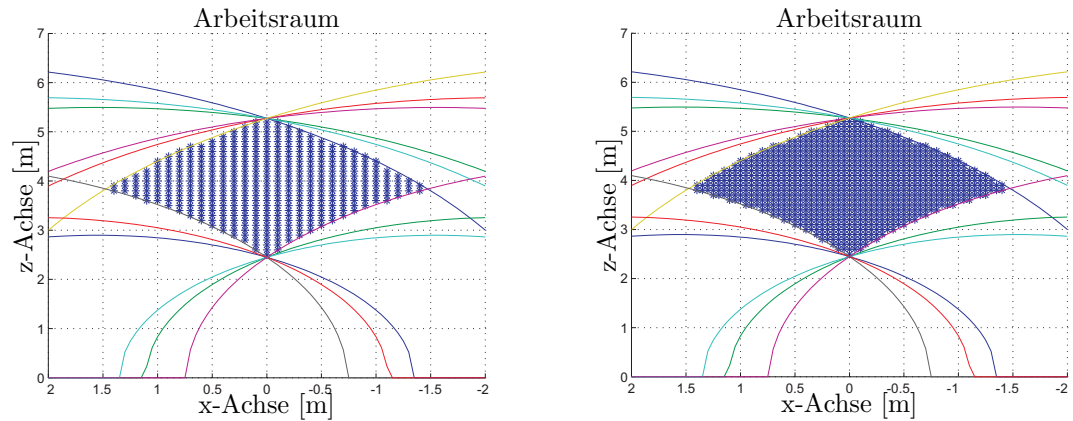


Bild 3.6: Vergleich der diskreten und geometrischen Untersuchungsmethode zur Arbeitsraumberechnung in der xz -Ebene; Links: diskrete Unterteilung 0.1 m; Rechts: diskrete Unterteilung 0.05 m

In Bild 3.6 ist der COW für die diskrete und die geometrische Untersuchungsmethode in der xz -Ebene bei $y = 0$ überlagert dargestellt. Es sind deutlich die Unterschiede der beiden Methoden zu erkennen. Während bei der geometrischen Methode nur die Begrenzung des Arbeitsraums dargestellt wird, ergibt sich im Prinzip die gleiche Form bei der diskreten Methode durch Überprüfung eines 3-D Raums. Je nachdem wie fein die Diskretisierung gewählt wird erhöht sich entsprechend die Genauigkeit und Berechnungszeit bei dieser Methode. Der Vergleich der beiden Untersuchungsmethoden zeigt eine recht gute Übereinstimmung, die schon mit einer relativ groben Diskretisierung erreicht werden kann.

In Bild 3.7 ist der COW für die diskrete und die geometrische Untersuchungsmethode in der xy -Ebene bei $z_{\text{ref}} = 3.95$ m überlagert dargestellt. Hierbei ist jedoch

bei einer Diskretisierungszahl von 0.1 m eine relativ ungenaue Übereinstimmung an der Berandung parallel zur y -Achse zu erkennen. Diese wird mit feinerer Diskretisierung besser angenähert, jedoch ist immer noch ein erkennbarer Zwischenraum vorhanden.

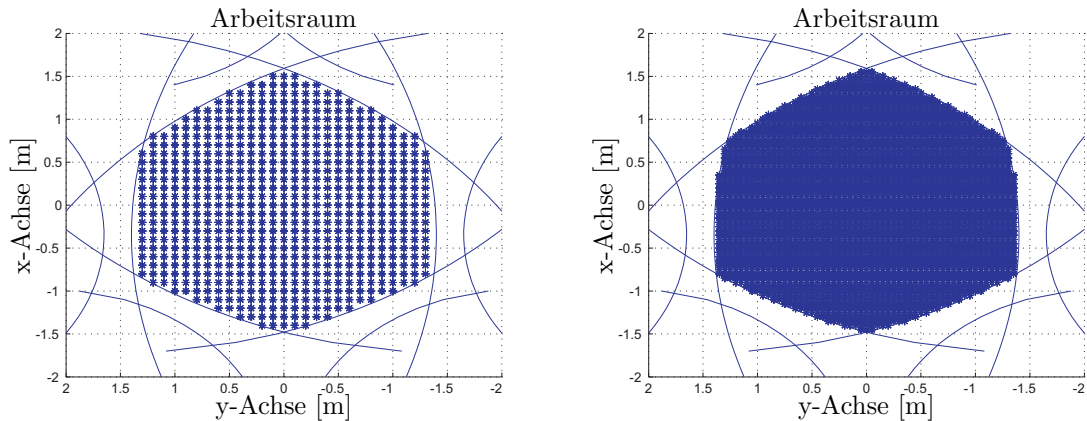


Bild 3.7: Vergleich der diskreten und geometrischen Untersuchungsmethode zur Arbeitsraumberechnung in der xy -Ebene; Links: diskrete Unterteilung 0.1 m; Rechts: diskrete Unterteilung 0.05 m

3.4 Abstandsberechnung zum Arbeitsraumrand

Die Berechnung des Abstandes zum Arbeitsraumrand ist für bestimmte Situationen während einer Simulationsfahrt sehr sinnvoll. Wird nun beispielsweise eine Kurve mit dem Fahrsimulator gefahren, die einen relativ großen Radius hat ist in Kenntnis zu setzen, ob die Plattform aus der derzeitigen Position genug Bewegungsmöglichkeiten hat, um diese Kurve zu durchfahren. Reicht der momentane Abstand zum Arbeitsraumrand für dieses Manöver nicht aus, kann darauf hin eine geeignete Vorpositionierung der Plattform eingeleitet werden um den Arbeitsraum effektiver nutzen zu können. Deshalb besteht ein Ziel dieser Arbeit darin aus der ermittelten Position der Plattform die Abstände zum Arbeitsraumrand zu bestimmen. Da die Berechnung während der Simulationsfahrt angewendet werden, soll diese in Echtzeit durchgeführt werden.

3.4.1 Berechnung für translatorische Abstände zum Arbeitsraumrand mit der diskreten Untersuchungsmethode

Die Hauptaufgabe des Algorithmus ist es, aus einer bestimmten Position der Plattform (vgl. Bild 3.8, roter Punkt) die Abstände zum Arbeitsraumrand zu berechnen. Dies soll in Echtzeit erfolgen.

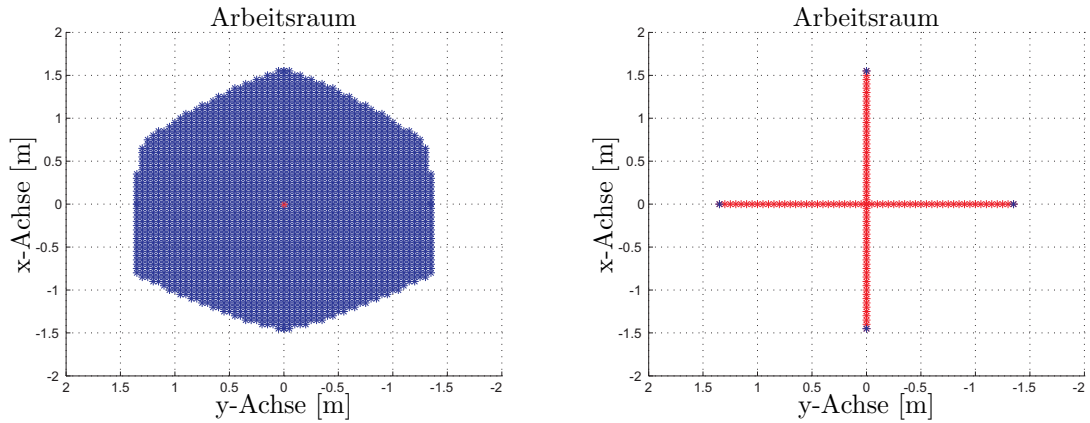
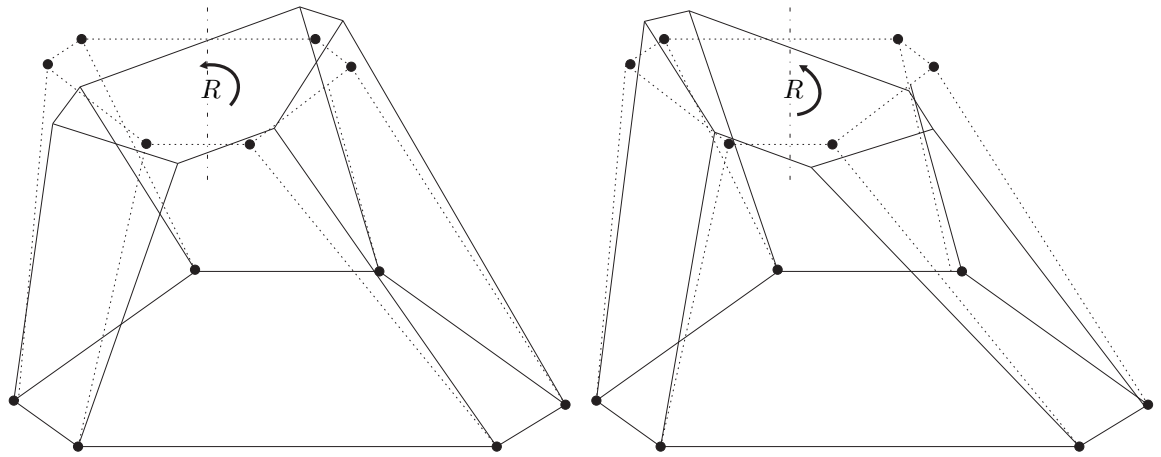


Bild 3.8: Arbeitsraum für eine konstante Höhe $z_i = z_{\text{ref}}$ mit vorgegebener konstanter Stellung $\varphi = \theta = \psi = 0^\circ$ der Plattform

In Bild 3.8 ist die prinzipielle Vorgehensweise zur Bestimmung des Abstandes zum Arbeitsraumrand dargestellt. Die Punkte, die auf der Linie der x - bzw. y -Koordinate liegen, werden ausgehend vom Referenzpunkt auf ihre Gültigkeit hin überprüft. Die Berechnung startet vom Referenzpunkt in alle vier translatorischen Richtungen. Tritt die Abbruchbedingung von Gleichung 2.14 aus Kapitel 2.5 ein, wird die Berechnung eingestellt und der letzte gültige Wert als Arbeitsraumrand bewertet. Dieser Wert wird von der Referenzposition subtrahiert, wodurch sich der Abstand zum Arbeitsraumrand ergibt.

3.4.2 Berechnung der Schwenkwinkel zum Arbeitsraumrand mit der diskreten Untersuchungsmethode

Für der Berechnung der maximal und minimal möglichen Schwenkwinkel der Plattform wird nach demselben Prinzip wie bei der Berechnung für translatorische Abstände vorgegangen. Hierbei sind die Eingangswerte für den Algorithmus der diskret unterteilte Wertebereich für die Winkel. Im Prinzip wird hier der OW berechnet. Von


 Bild 3.9: Schwenken der Plattform um einen Referenzpunkt R

einer Referenzposition, werden die maximal und minimal möglichen Schwenkwinkel berechnet (vgl. Bild 3.9), was gleichzeitig den Arbeitsraumrand für die möglichen Verdrehungen der Plattform ergibt.

3.4.3 Berechnung der horizontalen und vertikalen Abstände zum Arbeitsraumrand mit der geom. Untersuchungsmethode in der xy -Ebene

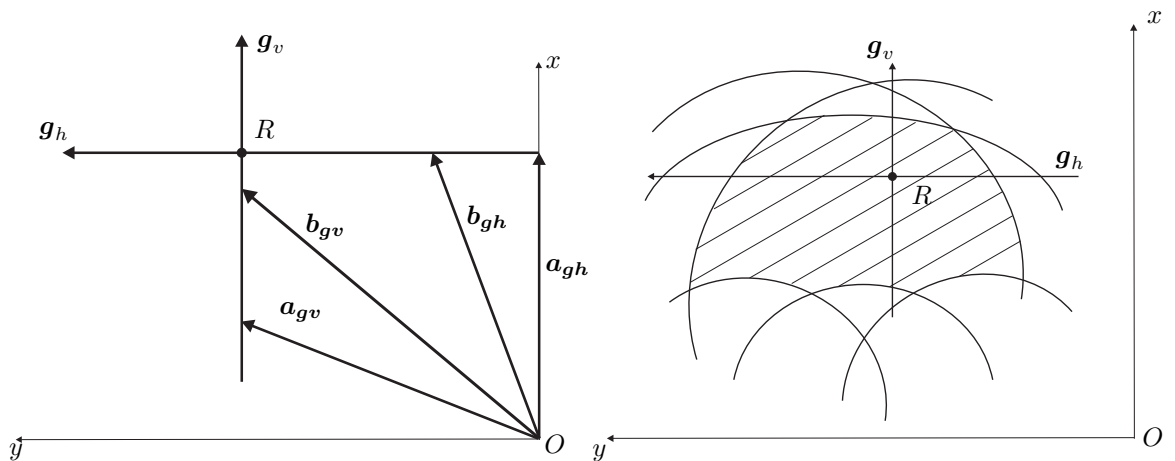


Bild 3.10: Geraden g_v und g_h ; Links: Gemeinsamer Schnittpunkt R in der xy -Ebene; Rechts: Abstandsberechnung in der xy -Ebene

Eine andere Vorgehensweise wird bei der Berechnung des Abstandes von einer Referenzposition zum Arbeitsraumrand eines COW wird mit der analytischen Untersuchungsmethode durchgeführt. Es wird ein 2-D Arbeitsraum für eine konstante Höhe z_h betrachtet. Für die Abstandsberechnung werden zunächst zwei Geraden

$$\mathbf{g}_h = \mathbf{a}_{gh} + m_h(\mathbf{b}_{gh} - \mathbf{a}_{gh}) \quad (3.1)$$

$$\mathbf{g}_v = \mathbf{a}_{gv} + m_v(\mathbf{b}_{gv} - \mathbf{a}_{gv}) \quad (3.2)$$

definiert, bei der die Vektoren $\mathbf{a}_{gk} = [a_{gkx} \ a_{gky} \ a_{gzk}]$ und $\mathbf{b}_{gk} = [b_{gkx} \ b_{gky} \ b_{gzk}]$ die Ortsvektoren darstellen. Hierbei unterscheidet $k \in [h, v]$ zwischen den Koordinaten der beiden Geraden. Die Vektoren \mathbf{a}_{gk} und \mathbf{b}_{gk} werden so gewählt, dass die Gerade \mathbf{g}_v parallel zur x -Achse und die Gerade \mathbf{g}_h parallel zur y -Achse verläuft (vgl. Bild 3.10, Links). Die beiden Geraden schneiden sich in einem Punkt R , der gleichzeitig auch die Koordinaten für die Plattform als momentanen Referenzpunkt angibt. Um die Schnittpunkte mathematisch zu bestimmen, werden die Geraden

$$\mathbf{g}_k = \begin{bmatrix} a_{gkx} + m_k(b_{gkx} - a_{gkx}) \\ a_{gky} + m_k(b_{gky} - a_{gky}) \\ a_{gzk} + m_k(b_{gzk} - a_{gzk}) \end{bmatrix} \quad (3.3)$$

mit den hergeleiteten Kugelgleichungen aus Kapitel 2.6, die sich für einen ebenen Arbeitsraum ($z_{\text{ref}} = \text{konstant}$) zu Kreisgleichungen reduzieren

$$\mathbf{x}_{\text{Kreis},j} = \begin{bmatrix} x_r \\ y_r \\ z_{rj} = \pm \sqrt{\rho_j^2 - (x_r - a'_x)^2 - (y_r - a'_y)^2} \end{bmatrix}, \quad (3.4)$$

gleichgesetzt:

$$\mathbf{x}_{\text{Kreis},j} = \mathbf{g}_k. \quad (3.5)$$

Dabei unterscheidet j zwischen den Kreisgleichungen mit minimalen ρ_{\min} und maximalen ρ_{\max} Radius. Es ergeben sich jeweils drei Gleichungen zu

$$x_r = a_{gkx} + m_k(b_{gkx} - a_{gkx}) \quad (3.6)$$

$$y_r = a_{gky} + m_k(b_{gky} - a_{gky}) \quad (3.7)$$

$$z_{r,j} = a_{gzk} + m_k(b_{gzk} - a_{gzk}). \quad (3.8)$$

Für die Gerade \mathbf{g}_v sind die y -Komponenten konstant ($a_{gyk} = b_{gyk}$), während bei der Geraden \mathbf{g}_h die x -Komponenten konstant sind ($a_{gxk} = b_{gxk}$). Da der Arbeitsraum für eine konstante Höhe z_h betrachtet wird, sind auch die z -Koordinaten der Geraden für diesen Fall immer gleich ($a_{gzk} = b_{gzk}$). Für die Berechnung der Schnittpunkte in x -Richtung wird durch Einsetzen von \mathbf{g}_v in 3.5 die Gleichung 3.8 unter Berücksichtigung der angegebenen Vereinfachungen nach

$$x_r = \pm \sqrt{\rho_j^2 - (a_{gzv} - a'_z)^2 - (y_r - a'_y)^2} + a'_x \quad (3.9)$$

aufgelöst. Zur Bestimmung der Schnittpunkte in y -Richtung wird \mathbf{g}_h in 3.5 und die Gleichung 3.8 nach

$$y_r = \pm \sqrt{\rho_j^2 - (a_{gzh} - a'_z)^2 - (x_r - a'_x)^2} + a'_y \quad (3.10)$$

aufgelöst. Zur Kontrolle können die Ergebnisse aus 3.9 und 3.10 in Gleichung 3.6 bzw. 3.7 eingesetzt und nach

$$m_h = \frac{(y_r - a_{gyh})}{(b_{gyh} - a_{gyh})} \quad (3.11)$$

$$m_v = \frac{(x_r - a_{gxv})}{(b_{gxv} - a_{gxv})} \quad (3.12)$$

umgestellt werden. Diese werden in die Geradengleichung 3.1 bzw. 3.2 eingesetzt. Der Abstand zwischen dem Referenzpunkt und den Schnittpunkt der beiden Geraden mit den Kreissegmenten (vgl. Bild 3.10, Rechts), die den Arbeitsraum eingrenzen, ergeben den Abstand zum Arbeitsraumrand. Weil die Abstandsberechnung analytisch durchgeführt wird, ist sie gegenüber der diskreten Methode genauer.

3.4.4 Simulationsergebnisse für die horizontale und vertikale Abstandsberechnung in der xy -Ebene

In Bild 3.11 sind die Simulationsergebnisse für die Schnittpunktberechnung in der xy -Ebene dargestellt. Es sind Schnittpunkte der Geraden mit den Ebenen dargestellt, die aber für die Abstandsberechnung keine Rolle spielen, weil sie außerhalb des Arbeitsraums liegen. Deshalb wird im nächsten Schritt das kürzeste Intervall bestimmt, was auch den Schnittpunkten der Geraden mit dem Arbeitsraumrand entspricht. Diese werden von dem Referenzpunkt subtrahiert, wodurch sich die Abstände zum Arbeitsraumrand ergeben.

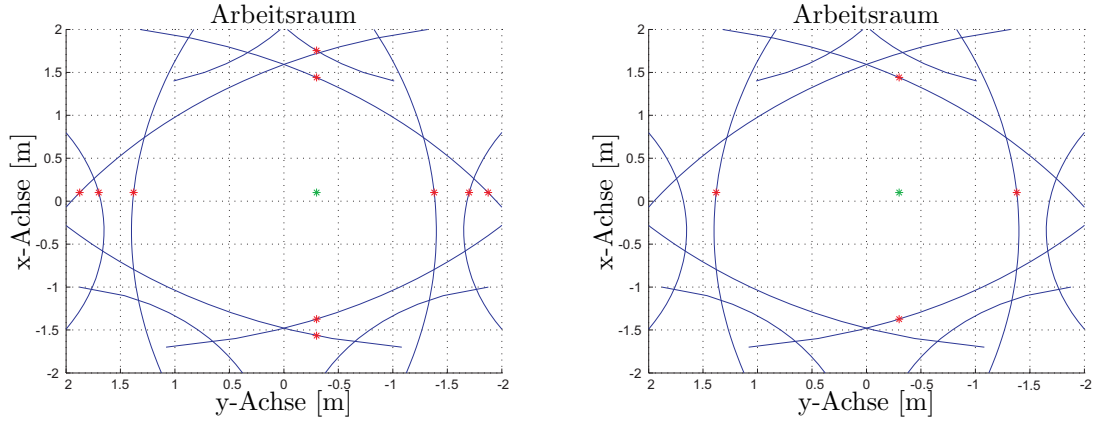


Bild 3.11: Abstandsbestimmung in der xy -Ebene; Links: Schnittpunkte; Rechts: Bestimmung des kürzesten Intervalls

3.4.5 Berechnung der horizontalen und vertikalen Abstände zum Arbeitsraumrand mit der geom. Untersuchungsmethode in der yz -Ebene

Bei der Abstandsberechnung in der yz -Ebene wird ebenfalls von einem 2-D Arbeitsraum ausgegangen ($x_h = \text{konstant}$). Für die Gerade parallel zur y -Achse wird analog zum vorigen Unterkapitel vorgegangen. Bei der Geraden parallel zur z -Achse:

$$\mathbf{l}_g = \mathbf{r}_g + m_z(\mathbf{t}_g - \mathbf{r}_g) \quad (3.13)$$

ergibt sich ein Unterschied bei der Umstellung der Variablen, wenn man die Gleichung mit den Kreisgleichungen 3.4

$$x_r = r_{gx} + m_z(t_{gx} - r_{gx}) \quad (3.14)$$

$$y_r = r_{gy} + m_z(t_{gy} - r_{gy}) \quad (3.15)$$

$$z_{r,j} = r_{gz} + m_z(t_{gz} - r_{gz}) \quad (3.16)$$

gleichsetzt. Da für diesen Fall $r_{gx} = t_{gx}$ und $r_{gy} = t_{gy}$ ist, ergibt sich aus Gleichung 3.14 $x_r = r_{gx}$ und 3.15 $y_r = r_{gy}$. Deshalb kann nur Gleichung 3.16 verwendet werden, und nach

$$m_z = \frac{\sqrt{\rho_j^2 - (r_{gz} - a'_x)^2 - (r_{gy} - a'_y)^2} - r_{gz}}{(t_{gz} - r_{gz})} \quad (3.17)$$

aufgelöst und in die Geradengleichung 3.13 eingesetzt werden. Dadurch ergeben sich die Schnittpunkte der Kreissegmente mit der Geraden \mathbf{l}_g .

3.4.6 Simulationsergebnisse für die horizontale und vertikale Abstandsberechnung in der yz -Ebene

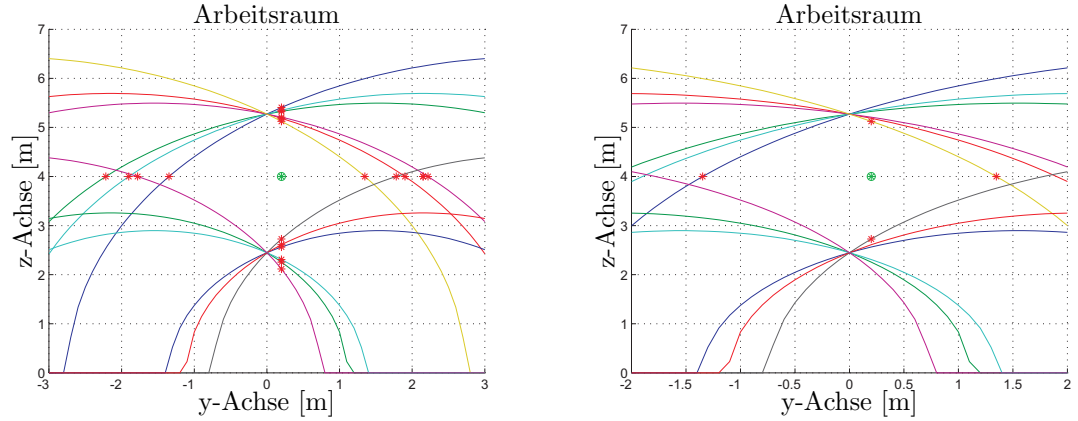


Bild 3.12: Abstandsbestimmung in der yz -Ebene; Links: Schnittpunkte; Rechts: Bestimmung des kürzesten Intervalls

In Bild 3.12 ist die Schnittpunktsberechnung in der yz -Ebene dargestellt. Auch hierbei wird das kürzeste Intervall an Schnittpunkten bestimmt, mit denen wiederum der Abstand vom Referenzpunkt zum Arbeitsraumrand bestimmt werden kann.

3.5 Vergleich der beiden Untersuchungsmethoden

In diesem Abschnitt wird die Rechenzeit der beiden Untersuchungsmethoden zur Abstandsbestimmung aufgelistet und gegenübergestellt. Ein Vergleich soll zeigen inwieweit die Genauigkeit bei der diskreten Untersuchungsmethode Einfluss auf die Rechenzeit hat. Darüber hinaus wird die Genauigkeit der beiden Untersuchungsmethoden analysiert. Abschließend erfolgt eine Bewertung für ihre Anwendbarkeit am DLR-Fahrsimulator.

3.5.1 Auflistung der Berechnungszeit für die diskrete Untersuchungsmethode

Im folgenden werden zunächst die Rechenzeiten für die diskrete Untersuchungsmethode gemessen. Die Werte wurden offline für einer Höhe von $z_{\text{ref}} = 3.95$ m ermittelt. Da in dieser Ebene die größten Bewegungsmöglichkeiten bestehen, kann deshalb

auch von den längsten Berechnungszeiten ausgegangen werden. Deshalb geben die Berechnungen aus dieser Referenzposition die obere Grenze der Rechenzeit wieder. Die Messungen wurden mit einem Windows XP System durchgeführt, und insgesamt zehn mal wiederholt, da sich gezeigt hat, dass die Prozessorrechenzeit bei mehrfachen Durchrechnen leicht schwankt. Jedoch sind die Abweichungen nicht gravierend, so dass eine gewisse Regelmäßigkeit festgestellt und tendenzielle Aussagen getroffen werden können. Dazu werden der Mittelwert \bar{x} und die Standardabweichung σ berechnet. Die Rechenzeit wurde mit dem Profiler von MATLAB gemessen, welche die Berechnungszeit des Algorithmus (Total Time) ausgibt. Mit Self Time wird die Zeit der aufgerufenen Funktionen im Algorithmus ausgegeben.

| Diskrete Analyse in der xy -Ebene | | | | |
|-------------------------------------|----------------|---------------|----------------|---------------|
| Messung | Total Time [s] | Self Time [s] | Total Time [s] | Self Time [s] |
| 1 | 0.062 | 0.047 | 0.078 | 0.063 |
| 2 | 0.063 | 0.048 | 0.078 | 0.063 |
| 3 | 0.047 | 0.047 | 0.078 | 0.063 |
| 4 | 0.047 | 0.031 | 0.078 | 0.063 |
| 5 | 0.047 | 0.047 | 0.078 | 0.063 |
| 6 | 0.047 | 0.047 | 0.078 | 0.063 |
| 7 | 0.063 | 0.048 | 0.078 | 0.063 |
| 8 | 0.047 | 0.031 | 0.078 | 0.063 |
| 9 | 0.062 | 0.048 | 0.078 | 0.063 |
| 10 | 0.047 | 0.047 | 0.078 | 0.063 |
| \bar{x} | 0.0532 | | 0.078 | |
| σ | 0.008 | | 0 | |

Tabelle 3.1: Diskrete Unterteilung für die Abstandsberechnung; Links: 0.1 m; Rechts: 0.05 m

In Tabelle 3.1 sind die Rechenzeiten für die diskrete Untersuchungsmethode in der xy -Ebene aufgelistet. Diese wurden für eine Diskretisierung von 0.1 m und 0.05 m gemessen. Dabei ist deutlich zu erkennen, dass die Rechenzeit für eine Diskretisierung von 0.1 m mit 0.0532 s kürzer ist als mit einer feineren Diskretisierung, die 0.078 s andauert. Bei der Bestimmung der Rechenzeiten haben sich relativ kleine Standardabweichungen ergeben, was die Ergebnisse aussagekräftig macht.

| Diskrete Analyse in der yz -Ebene | | | | |
|-------------------------------------|----------------|---------------|----------------|---------------|
| Messung | Total Time [s] | Self Time [s] | Total Time [s] | Self Time [s] |
| 1 | 0.063 | 0.063 | 0.079 | 0.079 |
| 2 | 0.078 | 0.047 | 0.110 | 0.094 |
| 3 | 0.094 | 0.063 | 0.062 | 0.062 |
| 4 | 0.078 | 0.047 | 0.062 | 0.062 |
| 5 | 0.078 | 0.047 | 0.078 | 0.078 |
| 6 | 0.078 | 0.047 | 0.090 | 0.062 |
| 7 | 0.046 | 0.046 | 0.063 | 0.063 |
| 8 | 0.094 | 0.063 | 0.078 | 0.078 |
| 9 | 0.047 | 0.047 | 0.109 | 0.062 |
| 10 | 0.094 | 0.063 | 0.125 | 0.094 |
| \bar{x} | 0.075 | | 0.0856 | |
| σ | 0.0179 | | 0.0224 | |

Tabelle 3.2: Diskrete Unterteilung für die Abstandsberechnung; Links: 0.1 m; Rechts: 0.05 m

| Diskrete Analyse in der xz -Ebene | | | | |
|-------------------------------------|----------------|---------------|----------------|---------------|
| Messung | Total Time [s] | Self Time [s] | Total Time [s] | Self Time [s] |
| 1 | 0.093 | 0.047 | 0.156 | 0.094 |
| 2 | 0.098 | 0.062 | 0.063 | 0.063 |
| 3 | 0.047 | 0.047 | 0.078 | 0.078 |
| 4 | 0.094 | 0.078 | 0.144 | 0.079 |
| 5 | 0.047 | 0.047 | 0.094 | 0.094 |
| 6 | 0.110 | 0.063 | 0.078 | 0.078 |
| 7 | 0.046 | 0.046 | 0.078 | 0.078 |
| 8 | 0.094 | 0.062 | 0.140 | 0.078 |
| 9 | 0.047 | 0.047 | 0.156 | 0.094 |
| 10 | 0.094 | 0.047 | 0.141 | 0.095 |
| \bar{x} | 0.077 | | 0.1128 | |
| σ | 0.0265 | | 0.0376 | |

Tabelle 3.3: Diskrete Unterteilung für die Abstandsberechnung; Links: 0.1 m; Rechts: 0.05 m

In Tabelle 3.2 und 3.3 sind die Messungen für die yz - und xz -Ebene aufgelistet. Hierbei ist zu erkennen, dass bei einigen Messungen mit feinerer Diskretisierung kürzere Rechenzeiten vorweisen als mit gröberer Diskretisierung. Damit lassen sich auch die relativ großen Standardabweichungen erklären. Jedoch ist prinzipiell die gleiche Tendenz wie in Tabelle 3.1 erkennbar, dass mit höherer Genauigkeit eine längere Rechenzeit einhergeht, was der Mittelwert belegt. Die Abweichungen der einzelnen Messungen von dieser Tendenz können auf den belegten Speicher des Prozessors zurückgeführt werden.

3.5.2 Auflistung der Berechnungszeit der Schwenkwinkel mit der diskreten Untersuchungsmethode

Laut Tabelle 2.1 sind Schwenkwinkel mit der Plattform von bis zu $\pm 21^\circ$ möglich. Diese sind aber nicht von jeder Position aus erreichbar, da dies wiederum von dem zur Verfügung stehenden Hub der Aktuatoren abhängt. Jedoch wurde bei der Ermittlung der Berechnungszeit die Neutralposition ausgewählt, so dass auch hier die maximal möglichen Verfahrswege berücksichtigt sind. Bei abweichenden Referenzpositionen kann sich die Berechnungszeit entsprechend nur noch verkürzen.

In Tabelle 3.4 sind die Rechenzeiten für die Schwenkwinkel um die x -Achse aufgelistet. Mit einer Diskretisierung von 0.5° ist eine sehr kurze Rechenzeit von bis zu 0.016 s (vgl. Messung 1,2,10) möglich. Der Mittelwert beträgt 0.034 s was relativ günstig für die online-Berechnung ist. Die Rechenzeiten bei einer Diskretisierung von 0.1° liegen im Bereich von 0.0702 s und sind damit erheblich länger. Für die derzeitige Anforderung ist eine Genauigkeit von 0.5° völlig ausreichend, da eine kürzere Rechenzeit höhere Priorität hat als die Genauigkeit. Dieselben Ergebnisse lassen sich im Prinzip auch auf andere Schwenkwinkel aus Tabelle 3.5 und 3.6 übertragen.

| Diskrete Analyse für den Schwenkwinkel φ | | | | |
|--|----------------|---------------|----------------|---------------|
| Messung | Total Time [s] | Self Time [s] | Total Time [s] | Self Time [s] |
| 1 | 0.016 | 0.016 | 0.094 | 0.062 |
| 2 | 0.016 | 0.016 | 0.094 | 0.078 |
| 3 | 0.031 | 0.016 | 0.078 | 0.046 |
| 4 | 0.046 | 0.031 | 0.047 | 0.047 |
| 5 | 0.046 | 0.031 | 0.062 | 0.062 |
| 6 | 0.046 | 0.031 | 0.093 | 0.046 |
| 7 | 0.046 | 0.031 | 0.078 | 0.063 |
| 8 | 0.031 | 0.031 | 0.062 | 0.062 |
| 9 | 0.046 | 0.031 | 0.047 | 0.047 |
| 10 | 0.016 | 0.016 | 0.047 | 0.047 |
| \bar{x} | 0.034 | | 0.0702 | |
| σ | 0.0138 | | 0.0198 | |

Tabelle 3.4: Diskrete Unterteilung für den Schwenkwinkel φ ; Links: 0.5° ; Rechts: 0.1°

| Diskrete Analyse für den Schwenkwinkel θ | | | | |
|---|----------------|---------------|----------------|---------------|
| Messung | Total Time [s] | Self Time [s] | Total Time [s] | Self Time [s] |
| 1 | 0.047 | 0.047 | 0.063 | 0.063 |
| 2 | 0.016 | 0.016 | 0.062 | 0.062 |
| 3 | 0.047 | 0.031 | 0.094 | 0.063 |
| 4 | 0.016 | 0.016 | 0.094 | 0.063 |
| 5 | 0.047 | 0.031 | 0.078 | 0.031 |
| 6 | 0.047 | 0.047 | 0.078 | 0.031 |
| 7 | 0.047 | 0.047 | 0.062 | 0.062 |
| 8 | 0.016 | 0.016 | 0.063 | 0.063 |
| 9 | 0.016 | 0.016 | 0.093 | 0.047 |
| 10 | 0.031 | 0.031 | 0.063 | 0.063 |
| \bar{x} | 0.033 | | 0.075 | |
| σ | 0.0154 | | 0.0143 | |

Tabelle 3.5: Diskrete Unterteilung für den Schwenkwinkel θ ; Links: 0.5° ; Rechts: 0.1°

| Diskrete Analyse für den Schwenkwinkel ψ | | | | |
|---|----------------|---------------|----------------|---------------|
| Messung | Total Time [s] | Self Time [s] | Total Time [s] | Self Time [s] |
| 1 | 0.047 | 0.047 | 0.078 | 0.032 |
| 2 | 0.047 | 0.016 | 0.062 | 0.062 |
| 3 | 0.047 | 0.031 | 0.047 | 0.047 |
| 4 | 0.047 | 0.016 | 0.078 | 0.047 |
| 5 | 0.031 | 0.031 | 0.078 | 0.046 |
| 6 | 0.047 | 0.047 | 0.078 | 0.047 |
| 7 | 0.047 | 0.047 | 0.047 | 0.047 |
| 8 | 0.031 | 0.016 | 0.062 | 0.062 |
| 9 | 0.016 | 0.016 | 0.047 | 0.047 |
| 10 | 0.016 | 0.031 | 0.062 | 0.062 |
| \bar{x} | 0.0376 | | 0.0639 | |
| σ | 0.0131 | | 0.0136 | |

Tabelle 3.6: Diskrete Unterteilung für den Schwenkwinkel ψ ; Links: 0.5° ; Rechts: 0.1°

3.5.3 Auflistung der Berechnungszeit für die geometrische Untersuchungsmethode

Bei der geometrischen Untersuchungsmethode ist eine Berechnung nur für translatorische Abstände (COW) möglich. Bei einer Neigung der Plattform würden sich die Koordinaten der transformierten Gelenke sich nicht mehr im Plattform-Mittelpunkt treffen, was jedoch bei der Untersuchungsmethode notwendig ist.

In Tabelle 3.7 sind die Rechenzeiten für die Abstandsberechnung in der xy -Ebene aufgelistet. Diese sind bei der geometrischen Untersuchungsmethode kürzer als im Vergleich zur diskreten Untersuchungsmethode. Jedoch ist der Unterschied für eine diskrete Unterteilung von 0.1 m relativ gering (vgl. Tabelle 3.1, $\bar{x} = 0.0532$ s mit Tabelle 3.7, $\bar{x} = 0.0343$ s). Diese Differenz nimmt selbstverständlich mit feinerer Diskretisierung zu. Der Vorteil der genaueren Ergebnisse mit der geometrischen Untersuchungsmethode wird im Gegensatz zur diskreten Untersuchungsmethode nicht mit einer längeren Rechenzeit erkaufte. Die in Tabelle 3.8 gemessenen Ergebnisse für die yz -Ebene sollten sich im Rahmen von Tabelle 3.7 orientieren. Die relativ lange Rechenzeiten sind auf die Prozessorrechenzeit zurückzuführen.

| Geometrische Analyse in der xy -Ebene | | |
|---|----------------|---------------|
| Messung | Total Time [s] | Self Time [s] |
| 1 | 0.046 | 0.046 |
| 2 | 0.031 | 0.031 |
| 3 | 0.032 | 0.032 |
| 4 | 0.031 | 0.031 |
| 5 | 0.031 | 0.031 |
| 6 | 0.031 | 0.031 |
| 7 | 0.032 | 0.032 |
| 8 | 0.047 | 0.047 |
| 9 | 0.031 | 0.016 |
| 10 | 0.031 | 0.016 |
| \bar{x} | 0.0343 | |
| σ | 0.0064 | |

Tabelle 3.7: Berechnungszeit nach der geometrischen Untersuchungsmethode für translatorische Abstände

| Geometrische Analyse in der yz -Ebene | | |
|---|----------------|---------------|
| Messung | Total Time [s] | Self Time [s] |
| 1 | 0.094 | 0.094 |
| 2 | 0.062 | 0.062 |
| 3 | 0.047 | 0.031 |
| 4 | 0.063 | 0.063 |
| 5 | 0.093 | 0.093 |
| 6 | 0.062 | 0.062 |
| 7 | 0.062 | 0.062 |
| 8 | 0.062 | 0.062 |
| 9 | 0.047 | 0.047 |
| 10 | 0.046 | 0.046 |
| \bar{x} | 0.0638 | |
| σ | 0.0172 | |

Tabelle 3.8: Berechnungszeit nach der geometrischen Untersuchungsmethode für translatorische Abstände

3.5.4 Bewertung für die Anwendung am Fahrsimulator

Es ist eindeutig erkennbar, dass die geometrische Untersuchungsmethode weniger Rechenzeit benötigt und dabei gleichzeitig genauere Ergebnisse liefert als die diskrete Untersuchungsmethode. Lediglich bei einer diskreten Unterteilung von 0.1 m ist die Berechnungsdauer fast genauso schnell wie die der geometrischen Untersuchungsmethode. Da die Anforderung der Aufgabenstellung möglichst in kurzer Zeit Ergebnisse zu erzielen ist, wobei die Genauigkeit nicht so sehr im Vordergrund steht, kann die diskrete Untersuchungsmethode für Versuchsfahrten online angewendet werden. Jedoch ist es auf lange Sicht betrachtet vorteilhafter die geometrische Untersuchungsmethode anzuwenden, da unabhängig von der ohnehin schon kürzeren Berechnungszeit im Prinzip immer von genauen Ergebnissen ausgegangen werden kann. Für die translatorischen Abstände zum Arbeitsraumrand bietet das geometrische Verfahren im Prinzip nur Vorteile gegenüber der diskreten Untersuchungsmethode.

Die Berechnungszeiten für die Schwenkwinkel mit der diskreten Untersuchungsmethode sind abhängig von der Genauigkeit völlig ausreichend bis sehr kurz. Mit einer diskreten Unterteilung von 0.5° kann eine relativ aussagekräftige Genauigkeit erzielt werden, womit eine Anwendung am Fahrsimulator zur Bestimmung der Schwenkwinkel sinnvoll ist. Mit dem geometrischen Verfahren sind bisher noch keine Berechnungen für den OW möglich.

4 Funktionstest am DLR-Fahrsimulator

In diesem Kapitel werden die Algorithmen auf ihre Funktionsweise am DLR-Fahrsimulator getestet. Die Eingangswerte für die Algorithmen werden aus einer beliebigen Simulationsfahrt geladen. Ziel ist es während der Simulationsfahrt möglichst viele Informationen über die Abstände von der aktuellen Position der Plattform zum Arbeitsraumrand zu erhalten.

4.1 Auswertung der translatorischen Bewegungen der Plattform mit der diskreten Untersuchungsmethode

Bei der diskreten Untersuchungsmethode hängt die Genauigkeit der Ergebnisse zur Berechnung der Abstände zum Arbeitsraumrand von der Schrittweite der Diskretisierung ab. Da mit groberer Diskretisierung die Rechenzeit sinkt aber auch gleichzeitig die Genauigkeit, gilt es hierbei den besten Kompromiss zwischen möglichst vielen Daten während der Fahrt und ausreichender Genauigkeit zu finden.

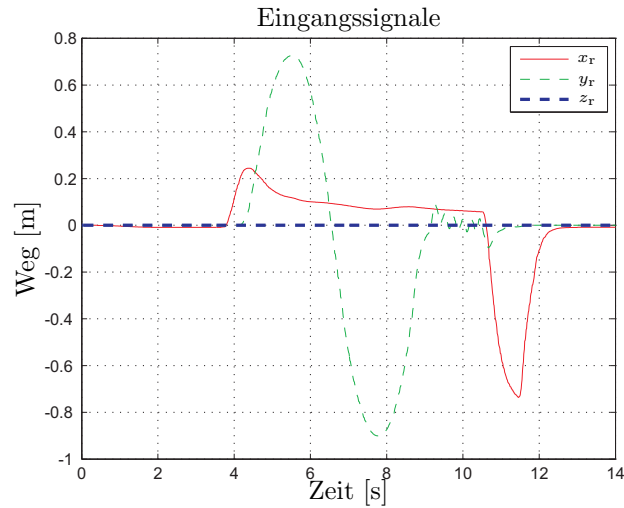


Bild 4.1: Eingangssignale in $x-y-z$ -Richtung vom Fahrsimulator für translatorische Bewegungen der Plattform

In Bild 4.1 sind die Eingangswerte für die translatorischen Bewegungen ($\varphi = \theta = \psi = 0^\circ$) der Plattform über die Zeit dargestellt. Die Eingangswerte dienen zur Bestimmung

der Ausgangsposition der Plattform für die Abstandsberechnung. Da die Simulationsfahrt für einer konstanten Höhe $z_i = z_{\text{ref}}$ von einer Neutralposition (x_{ref} y_{ref} z_{ref}) durchgeführt wird, bleibt die z_{ref} -Kooordinate konstant.

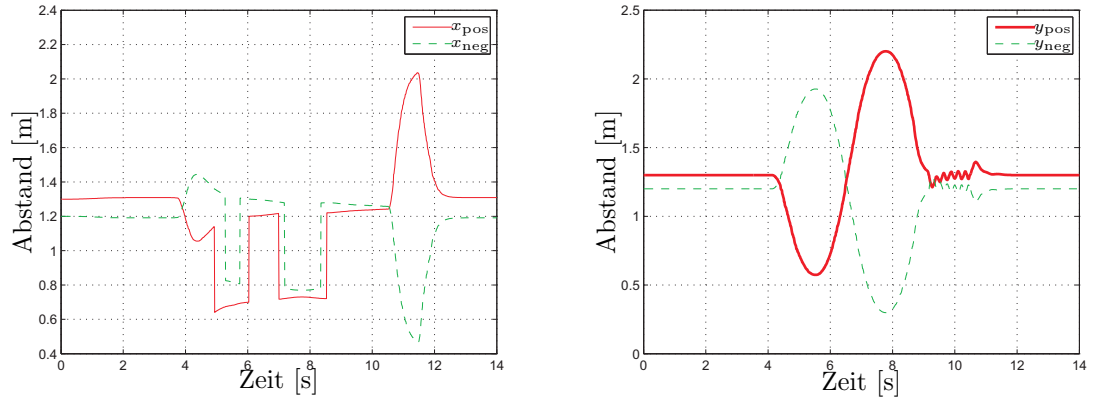


Bild 4.2: Ausgangssignal vom Fahrsimulator für eine Diskretisierung von 0.5 m; Links: Betrag der Abstände in positiver und negativer x -Richtung; Rechts: Betrag der Abstände in positiver und negativer y -Richtung

In Bild 4.2 sind die Beträge der positiven und negativen Abstände der Plattform zum Arbeitsraumrand in x - und y -Richtung über die Zeit dargestellt. Der Startwert gibt die Abstände ausgehend von der Neutralposition an. Es ist erkennbar, dass sobald die Plattform in positiver y -Richtung verfährt (vgl. Bild 4.1), entsprechend der Abstand zum Arbeitsraumrand in positiver y -Richtung abnimmt (vgl. Bild 4.2, y_{pos}). Im Gegensatz dazu nimmt der Abstand in positiver y -Richtung bei einer Bewegung der Plattform in negativer y -Richtung zu. Mit der Bewegung der Plattform in y -Richtung verringert sich gleichzeitig auch die Bewegungsmöglichkeit in beide x -Richtung (vgl. Bild 4.2, x_{pos} , x_{neg}), weshalb sich der Abstand zum Arbeitsraumrand verringert.

In Bild 4.3 sind die positiven Abstände in x - und y -Richtung mit einer Diskretisierung von 0.1 m über die Zeit aufgetragen. Dadurch ist vor allem im Bereich zwischen 4 und 8 s ein detaillierterer Verlauf der x -Abstände als im Vergleich zu Bild 4.2 dargestellt. Das liegt an der feineren Diskretisierung wodurch eine bessere Genauigkeit bei der Abstandsberechnung erreicht wird. Dies ist auch an dem Spitzenwert des x -Absandes zwischen 10 und 12 s erkennbar, weil hier ein Abstand von etwa 2.2 m

ermittelt wird, während in Bild 4.2 nur ca. 2 m angezeigt werden. Insgesamt wird bis zu 0.1 m auf beiden Seiten mehr zur Verfügung stehender Arbeitsraum ermittelt.

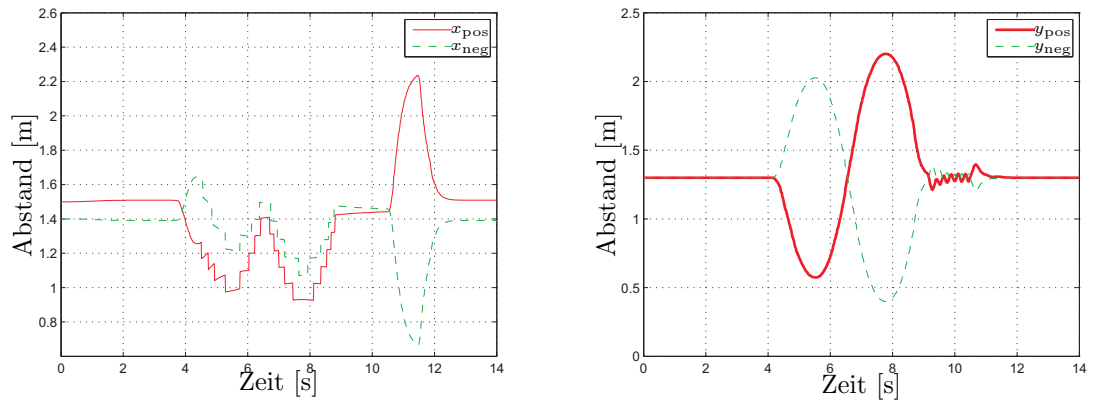


Bild 4.3: Ausgangssignal vom Fahrsimulator für eine Diskretisierung von 0.1 m; Links: Betrag der Abstände in positiver und negativer x -Richtung; Rechts: Betrag der Abstände in positiver und negativer y -Richtung

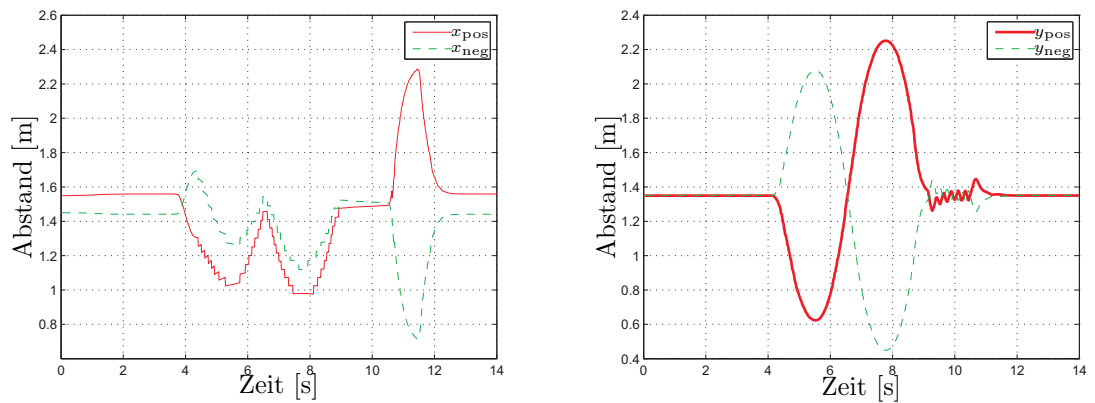


Bild 4.4: Ausgangssignal vom Fahrsimulator für eine Diskretisierung von 0.05 m; Links: Betrag der Abstände in positiver und negativer x -Richtung; Rechts: Betrag der Abstände in positiver und negativer y -Richtung

In Bild 4.4 ist dieselbe Berechnung für eine Diskretisierung von 0.05 m durchgeführt

worden. Hierbei sind nochmals genauere Werte ermittelt worden, die jedoch im Vergleich zu Bild 4.3 in positiver x -Richtung nicht so gravierend ausfallen. Die Differenz für x_{pos} im Vergleich zu Bild 4.3 beträgt zwischen 11 und 12 s ca. 0.05 m. Jedoch ist für die y -Richtung im Bereich zwischen 4 bis 6 s ist eine Differenz von mehr als 0.1 m zu erkennen. Trotz dieser höheren Genauigkeit ist der Algorithmus ausreichend schnell um genug Informationen auszugeben.

4.1.1 Funktionstest für translatorische Bewegungen der Plattform mit der geometrischen Untersuchungsmethode

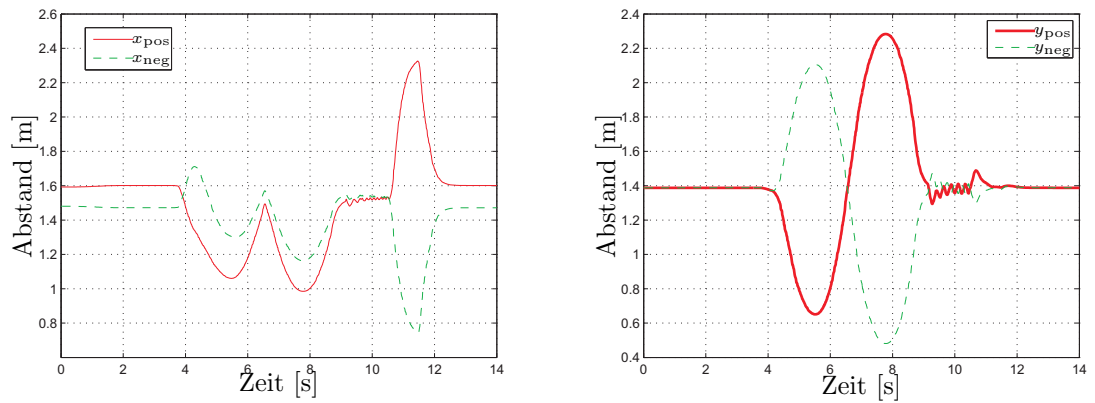


Bild 4.5: Ausgangssignal vom Fahrsimulator mit der geometrischen Untersuchungsmethode; Links: Betrag der Abstände in positiver und negativer x -Richtung; Rechts: Betrag der Abstände in positiver und negativer y -Richtung

In Bild 4.5 sind die Abstandsberechnungen mit der geometrischen Untersuchungsmethode durchgeführt worden. Hierbei kann aufgrund der analytischen Berechnung davon ausgegangen werden, dass es sich um die genauesten Ergebnisse handelt. Bedingt durch die genaue Berechnung der Methode ergeben sich vor allem entlang der positiven x -Richtung zwischen 4 bis 9 s keine stufenförmigen Annäherungen, sondern ein harmonischer Verlauf.

4.1.2 Vergleich der diskreten mit der geometrischen Untersuchungsmethode

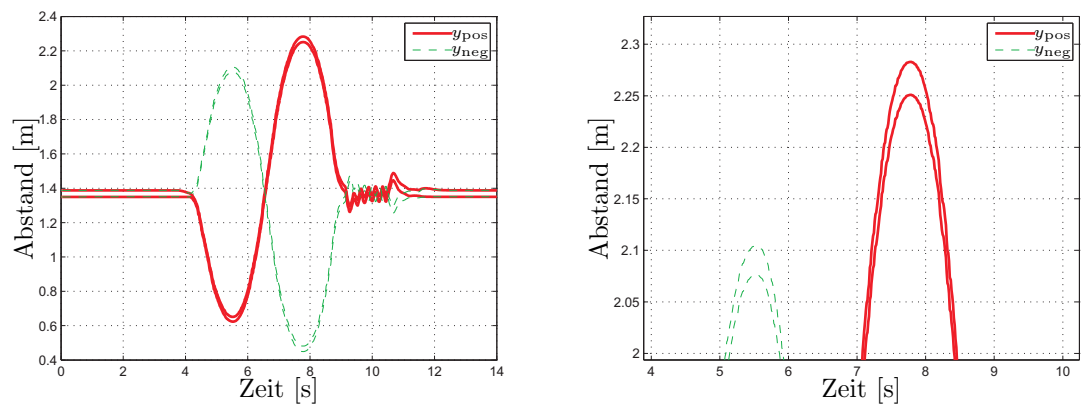


Bild 4.6: Überlagerung der Abstände der diskreten Untersuchungsmethode mit 0.05 m und der geometrischen Untersuchungsmethode; Links: gesamter Verlauf; Rechts: Ausschnitt bei 4 bis 10 s

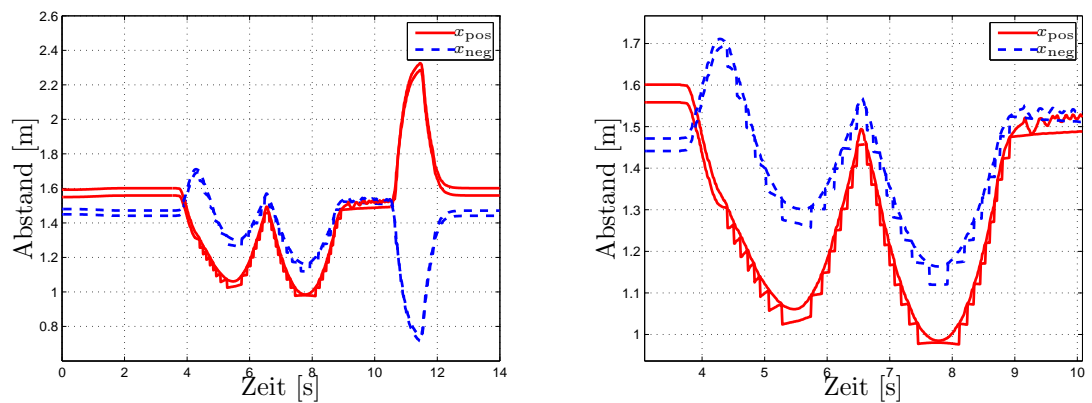


Bild 4.7: Überlagerung der Abstände der diskreten Untersuchungsmethode mit 0.05 m und der geometrischen Untersuchungsmethode; Links: gesamter Verlauf; Rechts: Ausschnitt bei 4 bis 9 s

In Bild 4.6 ist eine Überlagerung der beiden Untersuchungsmethoden für den Abstand in y -Richtung dargestellt. Ein Vergleich zeigt dass lediglich im Bereich zwischen 5 bis 6 und 7 bis 9 s eine geringe Abweichung für den y -Abstand vorhanden ist. Hier

zeigt die geometrische Untersuchungsmethode einen geringfügig höheren Wert als die diskrete Untersuchungsmethode an.

In Bild 4.7 ist eine Überlagerung der beiden Untersuchungsmethoden für den Abstand in x -Richtung dargestellt. Hierbei wird nochmals deutlich, dass abhängig von der Schrittweite der Diskretisierung immer besser angenähert wird. Die stufenförmige Annäherung bei der diskreten Methode tritt nur auf, wenn sich die Plattform gleichzeitig in y -Richtung bewegt (vgl. Bild 4.7, 4 bis 9 s). Eine Vermutung ist, dass durch die Bewegung in beide Richtungen gleichzeitig das Raster ungleichmäßig überprüft wird und es deshalb zu den stufenförmigen Abweichungen kommt. Diese nehmen jedoch mit feinerer Diskretisierung ab. Zur genaueren Untersuchung dieses Verlaufes müsste der Simulator in mehrere verschiedene Richtungen der xy -Ebene gesteuert werden, was jedoch aufgrund der derzeitigen Software-Konfiguration noch nicht möglich ist.

4.1.3 Funktionstest für Schwenkwinkel der Plattform

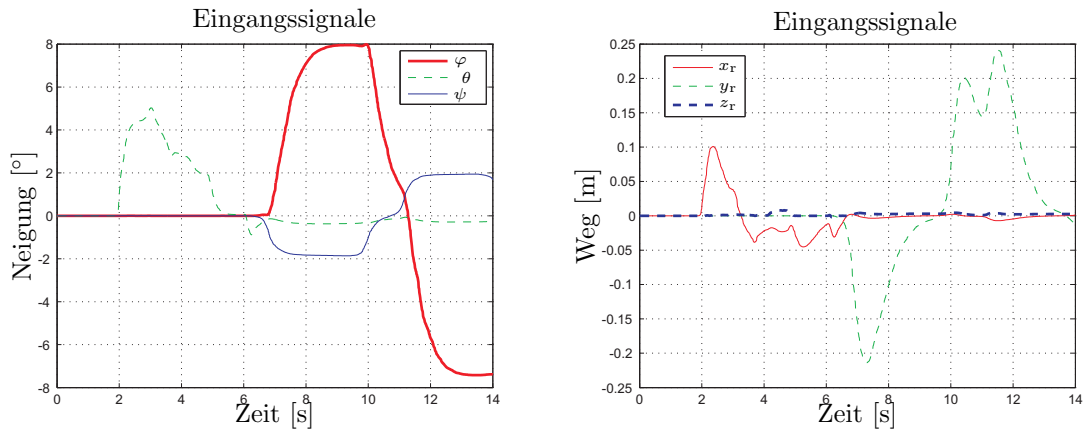


Bild 4.8: Eingangssignale vom Fahrsimulator für Neigung der Plattform; Links: Schwenkwinkel (φ, θ, ψ); Rechts: $x - y - z$ -Richtung

In Bild 4.8 sind die Eingangssignale für eine rotatorische (links) und translatorische (rechts) Bewegung der Plattform während einer Simulationsfahrt dargestellt. Nach der derzeitigen Software-Konfiguration ist es bisher nicht möglich aus einer konstanten Positionen die maximal möglichen Schwenkwinkel für einen Winkel isoliert überprüfen zu können. Deshalb kann hierbei von Abweichungen ausgegangen werden, weil

bedingt durch die zusätzlichen rotatorischen und translatorischen Bewegungen der Plattform die Voraussetzungen unter denen der Algorithmus arbeitet nicht mehr exakt erfüllt sind.

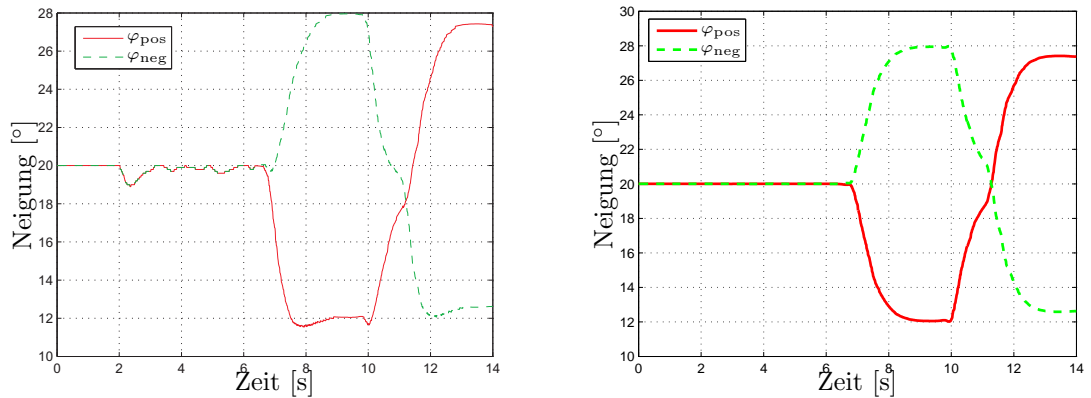


Bild 4.9: Ausgangssignal vom Fahrsimulator für Neigung φ der Plattform; Links: Ausgangssignal mit Berücksichtigung der translatorischen Bewegung; Rechts: Schwenkwinkel aus konstanter Neutralposition

In Bild 4.9 sind die positiven und negativen Abstände der Neigung der Plattform zum Arbeitsraumrand dargestellt. Der zu erwartende gegenläufige Verlauf des Abstandes zur Neigung der Plattform wird im Bereich zwischen 8 bis 10 s gestört (vgl. Bild 4.9, Links), was an der zusätzlichen translatorischen Bewegung der Plattform zu diesen Zeitpunkt (vgl. Bild 4.8) liegt. Die dadurch bedingt voreingestellten Aktuatoren können deshalb weniger Hub zur Verfügung stellen als von der Neutralposition aus betrachtet. Werden die translatorischen Eingangswerte unterdrückt und nur die rotatorischen betrachtet, ergibt sich eine bessere Übereinstimmung (vgl. Bild 4.9, Rechts) trotz Einfluss der Rotation der Plattform um die anderen beiden Achsen. Das liegt daran, weil die Abstände relativ weit von den maximal möglichen Schwenkwinkel von $\pm 20^\circ$ entfernt sind. Somit fällt die Ungenauigkeit nicht so stark ins Gewicht, was sich jedoch in Nähe des Arbeitsraumrandes gravierender bemerkbar machen würde.

4.2 Vergleich des translatorischen Arbeitsraums

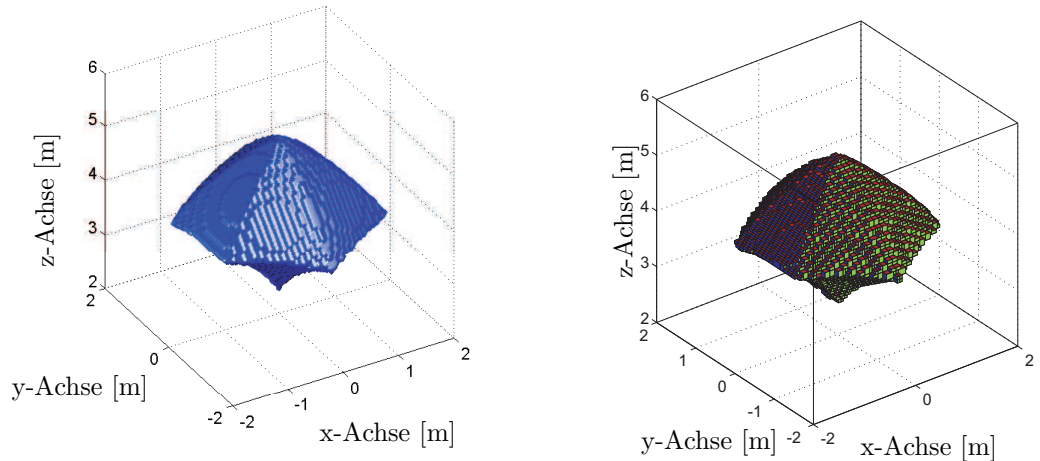


Bild 4.10: Translatorischer Arbeitsraum des Fahrsimulators ($\varphi = \theta = \psi = 0^\circ$); Links: Diskrete Unterteilung von 0.1 m; Rechts: Kontinuierlicher Arbeitsraum

In Bild 4.10 ist ein Vergleich des 3-D Arbeitsraums mit der diskreten Untersuchungsmethode und eines kontinuierlichen Arbeitsraums dargestellt. Durch Anwendung der Intervallanalyse ergeben sich Würfel, mit denen sich der Arbeitsraum einer PKM darstellen lässt [4]. Der Vorteil dieser Methode gegenüber der diskreten Methode ist, dass die Gesamtheit der Würfel sicher im Arbeitsraum enthalten sind, unabhängig von numerischen Rundungsfehlern. Die Genauigkeit liegt bei $\epsilon = 0.025$, was der kleinsten Kantenlänge der Würfel entspricht. Die Übereinstimmung der beiden graphischen Darstellungen ist relativ gut, was zum einen an der charakteristischen Gestalt und an den Abmaßen der beiden schirmähnlichen Formen erkennbar ist.

Da bei der derzeitigen Software-Konfiguration der 3-D Arbeitsraum am Fahrsimulator nicht validiert werden kann, ist der Vergleich zu einer unabhängigen Berechnungsmethode [4] die beste Möglichkeit die eigenen Berechnungen zu verifizieren.

4.3 Fazit

Anhand der Testergebnisse wird nochmals deutlich, dass mit einer höheren Diskretisierung auch eine bessere Genauigkeit erzielt werden kann. Gleichzeitig kann davon

ausgegangen werden, dass auch der Algorithmus mit der höchsten Diskretisierung ausreichend Informationen während der Simulationsfahrt liefert. Diese sind annähernd so gut wie bei der geometrischen Untersuchungsmethode (vgl. Bild 4.6). Mit einer noch höheren Diskretisierung (z. B. 0.01 m) kann eine noch bessere Annäherung erreicht werden, jedoch wird sich dies bei der Rechenzeit bemerkbar machen. Im Gegensatz dazu kann bei der geometrischen Untersuchungsmethode unabhängig von der Genauigkeit immer von einer kürzeren Rechenzeit ausgegangen werden.

5 Zusammenfassung und Ausblick

In dieser Arbeit werden zwei Untersuchungsmethoden zur Arbeitsraumberechnung des DLR-Fahrsimulators vorgestellt, die auf den kinematischen Eigenschaften einer PKM basieren. Anschließend wird der Arbeitsraum graphisch im 3-D Raum präsentiert. Da die Hauptbewegungsrichtungen des Simulators in einer Ebene stattfinden wird der ebener Arbeitsraum für verschiedene Orientierungen der Plattform im 2-D Raum dargestellt. Im nächsten Schritt wird ausgehend von einer Referenzposition der Plattform mit beiden Untersuchungsmethoden eine Abstandsberechnung zum Arbeitsraumrand durchgeführt. Um diese auf ihre Effizienz beurteilen zu können, werden die verschiedenen Algorithmen hinsichtlich ihrer Rechenzeit miteinander verglichen. Dadurch sollen während einer Simulationsfahrt möglichst viele Daten über den momentanen Abstand der Plattform zum Arbeitsraumrand geliefert werden. Die Ergebnisse der Funktionstests der Algorithmen, welche online am Fahrsimulator durchgeführt wurden, werden hinsichtlich ihrer Genauigkeit und Schnelligkeit ausgewertet und miteinander verglichen.

Die Funktionstests am DLR-Fahrsimulator haben gezeigt, dass im Prinzip ausreichend Daten zur Abstandsberechnung mit beiden Untersuchungsmethoden aufgrund der relativ kurzen Rechenzeit der Algorithmen möglich ist. Wobei die Ergebnisse zeigen, dass die geometrische Untersuchungsmethode eine genauere Berechnung bei gleichzeitig kürzerer Rechenzeit gegenüber der diskreten Methode erlaubt.

Die von der Abstandsberechnung erhaltenen Ergebnisse können dann als Basisinformation zur Ansteuerung des Fahrsimulators dienen, um eine geeignete Vorpositionierung einzuleiten. Somit könnte der zur Verfügung stehende Arbeitsraum effektiver genutzt werden.

Es können weitere Optimierungsverfahren durchgeführt werden, wie z. B. den Code in einer anderen Programmiersprache umzuschreiben um die Rechenzeit weiter zu verkürzen. Darüber hinaus können effizientere numerische Suchverfahren die Rechenzeit weiter verkürzen.

Da u. a. auch eine 3-D Modellierung des Arbeitsraums hergeleitet wird, kann diese für erweiterte Simulationsfahrten, wie z. B. eine Geländefahrt mit großen Winkel und vertikalen Bewegungen des Fahrsimulators als Grundlage zur Abstandsberechnung verwendet werden.

Eine Eigenschaft des Arbeitsraums ist die Symmetrie, die bei der Berechnung ausgenutzt werden kann. Da das System drei Symmetrieachsen hat, reicht es prinzipiell

aus nur ein sechstel des Arbeitsraums darzustellen und entsprechend die Abstandsberechnung durchzuführen. Dies sollte in zukünftigen Arbeiten genauer betrachtet werden.

Weiterhin sollte der Arbeitsraum auf mögliche Singularitäten untersucht werden. Der effektiv nutzbare Arbeitsraum wird durch Singularitäten eingeschränkt und macht deshalb eine Untersuchung notwendig. Verfährt der Fahrsimulator bei einer vorgegebenen Trajektorie dennoch in eine singuläre Stellung, kann die Mechanik des Simulators beschädigt werden.

Literatur

- [1] Gough, V. E.: *Contribution to discussion to papers on research in automobile stability and control and in tyre performance*, Proceedings of Auto, 1956-1957.
- [2] Stewart, D.: *A platform with six degrees of freedom*, proceedings of the institution of mechanical engineers, Vol. 180, No. 5, pp. 371-378, 1965.
- [3] Pott, A.: Entwicklung eines Dynamik-Simulationsmodells einer Parallel-Werkzeugmaschine. Diplomarbeit 2003, Lehrstuhl für Mechatronik, Universität Duisburg-Essen.
- [4] Pott, A.: Analyse und Synthese von Werkzeugmaschinen mit paralleler Kinetik. Dissertation vsl. 2007, Lehrstuhl für Mechatronik, Universität Duisburg-Essen.
- [5] Hiller, M., Pott, A.: *A Force Based Method for Complete Linearization of Complex Multibody Systems*, Advanced in Robotic Kinematics, KLAUER 2004.
- [6] Bayer, K.: Untersuchung eines Kalibrierverfahrens für Parallelkinematik-Werkzeugmaschinen. Studienarbeit 2005, Lehrstuhl für Mechatronik, Universität Duisburg-Essen.
- [7] Jean-Pierre Merlet: *Parallel Manipulators, Part 1*, INRIA UNITE DE RECHERCHE, Paris, 1987.
- [8] Jean-Pierre Merlet: *Parallel Robots*, KLUWER ACADEMIC PUBLISHER, Dordrecht / Boston / London, 2000.
- [9] Jean-Pierre Merlet: *Determination of the orientation workspace of parallel manipulators*, INRIA, Sophia-Antipolis, 1993
- [10] Jean-Pierre Merlet: *Determination of 6D Workspaces of Gough-Type Parallel Manipulator and Comparison between Different Geometries*, INRIA, Sophia-Antipolis, September 1999
- [11] Jean-Pierre Merlet: *Geometrical determination of the workspace of a constrained parallel manipulator*, ARK, Ferrare, 1992
- [12] Jean-Pierre Merlet: *Designing a parallel manipulator for a specific workspace*, INRIA, Sophia-Antipolis, April 1995

-
- [13] Jo D.Y., Haug E.J.: *Workspace analysis of closed loop mechanisms with unilateral constraints*, ASME Design Automation Conf., pages 53-60, Montreal, September 1989.
 - [14] Gosselin C.: *Determination of the workspace of 6-dof parallel manipulators.*, ASME J. of Mechanical Design, 112(3).377-382, June 1990.
 - [15] Fichter, E. F.: *A Stewart platform based manipulator: General theory and practical construction*, The international Journal of Robotics Research, Vol. 5, No. 3, pp. 157-181, 1986.
 - [16] Raghavan, M.: *The Stewart Platform of General Geometry Has 40 Configurations*, Power Systems Research Department, 30500 Mound Road.
 - [17] Hiller, M., Kecskemethy, A. und Woernle, C.: *Computergestützte Methoden und Verfahren - Vorlesungsskript*, Lehrstuhl für Mechatronik, Duisburg.
 - [18] Deutsches Zentrum für Luft- und Raumfahrt (DLR) - Institut für Verkehrsführung und Fahrzeugsteuerung (IFS): Automotive Systeme - Dynamischer Fahrsimulator http://www.dlr.de/fs/institut/abteilungen/automotive/index_4
 - [19] Conti, Joseph P.: *Workspace Variation of a hexapod machine tool*, Department of Mechanical Engineering & Institute for Systems Research, University of Maryland, USA
 - [20] Fink, B.: *Einführung in die Robotik - Vorlesungsskript*, Fachgebiet Mechatronik, Duisburg.

A Algorithmus für den diskreten 3-D COW

```
clc;
clear all;

% Minimale und maximale Aktuatorlänge
rho_min_limit = 4.45;
rho_max_limit = 6.45;

% Diskrete Unterteilung des kartesischen Raums in [m]: 0.1 0.05
x = -1.8:0.1:2;
y = -1.8:0.1:2;
z = 1.7:0.1:5.5;

% Definition der Orientierung der Plattform in [°]
phi = 0;
theta = 0;
psi = 0;

% Umrechnung von [°] in rad/s
phi = phi*pi/180;
theta = theta*pi/180;
psi = psi*pi/180;

% Leeres 6-dimensionales Feld für alle Freiheitsgrade
valid = zeros(length(x), length(y), length(z), length(phi),...
... length(theta), length(psi));

% Abstände der Gelenke (a_gelenk) zum Inertialsystem
% Abstände der Gelenke (b_gelenk) zum körperfesten
% Koordinatensystem an der Plattform
[a_gelenk,b_gelenk] = eingang_Gelenk_K0;

for i = 1:length(x)
    for j = 1:length(y)
        for k = 1:length(z)
            for l = 1:length(phi)
                for m = 1:length(theta)
```

```

    for n = 1:length(psi)
        % Berechnung der Aktuatorlänge für
        % vorgegebene Position
        [rho] = position_rho...
        ...([x(i), x(j), z(k), phi(1), theta(m),...
        ... psi(n)], [b_gelenk], [a_gelenk]);

        for p = 1:length(rho)

            % Untersuchung der Punkte für den Arbeitsraum,
            % durch Einhalten der technischen Limitierung
            if ((min(rho) > rho_min_limit) &&...
            ... (max(rho) < rho_max_limit))

                valid(j, i, k) = 1;

            end

        end

    end

end

end

end

end

end

end

end

end

% Graphische Darstellung
figure;
xlabel('x'), ylabel('y'), zlabel('z'), title('Arbeitsraum');
hold on;
grid on;

P=valid;

% Berechnung der Oberfläche aus dem berechneten Arbeitsraum-Volumen
% Parameter für beste Ergebnisse: 0.05 / 0.0001
isosf=isosurface(x,y,z,P,0.05);

% Farbe

```

```
p1=patch(isosf,'FaceColor','blue','EdgeColor','none');

% Abrundung der Oberflächenkanten
isonormals(x,y,z,P,p1);

% Gleichmäßige Achsen Auslegung
daspect([1 1 1])
view([-30 20]);

% Schattierung des Arbeitsraums
camlight; camlight(-80,-10);
lighting phong
```

B Algorithmus für den geometrischen 3-D COW

```
clc
clear all

rho_min = 4.45;

rho_max = 6.45;

ax = [3.57957 1.84752 -5.42709 -5.42709 1.84752 3.57957];
ay = [4.20000 5.20000 1.00000 -1.00000 -5.20000 -4.20000];

bx = [3.92598 -1.52998 -2.39600 -2.39600 -1.52998 3.92598];
by = [0.50000 3.65000 3.15000 -3.15000 -3.65000 -0.50000];

a_prime_x = ax - bx;
a_prime_y = ay - by;

x = -1.8:0.2:1.8;

% Für die Darstellung der Kugelebenen y = -1.8:0.2:1.8 !!!
% Für die Darstellung eines Schnitts durch die Kugelebenen
y = 0:0.2:1.8;

rx = [x]';
ry = [y]';
r = [x]';
[X,Y] = meshgrid(rx,ry);

% Berechnung der z Koordinaten der Kugelgleichungen
% / Auflösung nach z
z_max1 = sqrt(rho_max^2-(X-a_prime_x(1)).^2-(Y-a_prime_y(1)).^2);
z_min1 = sqrt(rho_min^2-(X-a_prime_x(1)).^2-(Y-a_prime_y(1)).^2);
z_max2 = sqrt(rho_max^2-(X-a_prime_x(2)).^2-(Y-a_prime_y(2)).^2);
z_min2 = sqrt(rho_min^2-(X-a_prime_x(2)).^2-(Y-a_prime_y(2)).^2);
z_max3 = sqrt(rho_max^2-(X-a_prime_x(3)).^2-(Y-a_prime_y(3)).^2);
z_min3 = sqrt(rho_min^2-(X-a_prime_x(3)).^2-(Y-a_prime_y(3)).^2);
z_max4 = sqrt(rho_max^2-(X-a_prime_x(4)).^2-(Y-a_prime_y(4)).^2);
```

```
z_min4 = sqrt(rho_min^2-(X-a_prime_x(4)).^2-(Y-a_prime_y(4)).^2);
z_max5 = sqrt(rho_max^2-(X-a_prime_x(5)).^2-(Y-a_prime_y(5)).^2);
z_min5 = sqrt(rho_min^2-(X-a_prime_x(5)).^2-(Y-a_prime_y(5)).^2);
z_max6 = sqrt(rho_max^2-(X-a_prime_x(6)).^2-(Y-a_prime_y(6)).^2);
z_min6 = sqrt(rho_min^2-(X-a_prime_x(6)).^2-(Y-a_prime_y(6)).^2);

figure;
xlabel('x'), ylabel('y'), zlabel('z'), title('Arbeitsraum');
axis([-2 2 -2 2]);
view([-40 20]);
hold on;
grid on

% Erstellen der Kugelebenen
mesh(X,Y,z_max1)
mesh(X,Y,z_min1)
mesh(X,Y,z_max2)
mesh(X,Y,z_min2)
mesh(X,Y,z_max3)
mesh(X,Y,z_min3)
mesh(X,Y,z_max4)
mesh(X,Y,z_min4)
mesh(X,Y,z_max5)
mesh(X,Y,z_min5)
mesh(X,Y,z_max6)
mesh(X,Y,z_min6)
```

C Berechnung der Abstände in der xy -Ebene mit der geometrischen Untersuchungsmethode

```
clc
clear all

% Minimale und maximale Aktuatorlänge
rho_min_limit = 4.45;
rho_max_limit = 6.45;

% Transformierte Gelenkkoordinaten
[a_prime_x,a_prime_y] = eingang_konst;

% Vorgabe des Referenzpunktes zur Abstandsberechnung
y_dist = 0;
x_dist = 0;

% Wertebereich für die Berechnung entlang der x-Achse
x = -2:0.1:2;
y = y_dist;
z = 3.95;

rz = z;
rx = [x]';
ry = [y]';

% Referenzpunkt der Plattform
R = [x_dist y_dist z]';

% Definition der Gerade parallel zur y-Achse,
% erzeugt aus dem Vektor agh und bgh
agx = x_dist;
agy = y_dist;
agz = z; % entspricht der Referenzhöhe der Plattform

% Ortsvektor der Geraden
agh = [agx agy agz];
```

```
bgx = agx;
bgy = 5;
bgz = z;

% Ortsvektor der Geraden
bgh = [bgx bgy bgz];

% Definition der Gerade parallel zur x-Achse,
% erzeugt aus dem Vektor agv und bgv
agx1 = x_dist;
agy1 = y_dist;
agz1 = z; % entspricht der Referenzhöhe der Plattform

agv = [agx1 agy1 agz1];

bgx1 = 5;
bgy1 = agy1;
bgz1 = agz1;

bgv = [bgx1 bgy1 bgz1];

% Speicherplatz eines Feldes für die x- und y-Koordinaten der Ebenen
yx_ref=cell(2,24);

for i =1:6

% Es werden imaginäre Werte berechnet, da Punkte außerhalb der Kreislinie
% berechnet werden.
% Deshalb werden nur die reellen Zahlen dargestellt
yr_max_p= a_prime_y(i) + sqrt(rho_max_limit^2...
... - (rz).^2 - (rx - a_prime_x(i)).^2);

yx_ref{1,(i-1)*4+1}=rx(imag(yr_max_p)==0);
yx_ref{2,(i-1)*4+1}= yr_max_p(imag(yr_max_p)==0);

yr_min_p= a_prime_y(i) + sqrt(rho_min_limit^2...
... - (rz).^2 - (rx - a_prime_x(i)).^2);
```

```
yx_ref{1,(i-1)*4+2}=rx(imag(yr_min_p)==0);
yx_ref{2,(i-1)*4+2}=yr_min_p(imag(yr_min_p)==0);

yr_max_n = a_prime_y(i) - sqrt(rho_max_limit^2...
... - (rz).^2 - (rx - a_prime_x(i)).^2);

yx_ref{1,(i-1)*4+3}=rx(imag(yr_max_n)==0);
yx_ref{2,(i-1)*4+3}=yr_max_n(imag(yr_max_n)==0);

yr_min_n = a_prime_y(i) - sqrt(rho_min_limit^2...
... - (rz).^2 - (rx - a_prime_x(i)).^2);

yx_ref{1,(i-1)*4+4}=rx(imag(yr_min_n)==0);
yx_ref{2,(i-1)*4+4}=yr_min_n(imag(yr_min_n)==0);

%%%Schnittpunkte der Geraden gv mit den Kreisebenen
xr_gerade_min_p(i) = (sqrt(-agz1^2 - (ry - a_prime_y(i))^2...
... + rho_min_limit^2) + a_prime_x(i));

xr_gerade_max_p(i) = (sqrt(-agz1^2 - (ry - a_prime_y(i))^2...
... + rho_max_limit^2) + a_prime_x(i));

xr_gerade_min_n(i) = (-sqrt(-agz1^2 - (ry - a_prime_y(i))^2...
... + rho_min_limit^2) + a_prime_x(i));

xr_gerade_max_n(i) = (-sqrt(-agz1^2 - (ry - a_prime_y(i))^2...
... + rho_max_limit^2) + a_prime_x(i));

% Berechnung der Schnittpunkte
m_min_p(i) = (xr_gerade_min_p(i) - agx1)/(bgx1-agx1);

m_max_p(i) = ( xr_gerade_max_p(i) - agx1)/(bgx1-agx1);

m_min_n(i) = (xr_gerade_min_n(i) - agx1)/(bgx1-agx1);

m_max_n(i) = ( xr_gerade_max_n(i) - agx1)/(bgx1-agx1);
```

```
% Einsetzen in Gerade
g_min_p(:,i) = [agx1 agy1 agz1]' + m_min_p(i)*...
...([bgx1 bgy1 bgz1]-[agx1 agy1 agz1])';

g_max_p(:,i) = [agx1 agy1 agz1]' + m_max_p(i)*...
...([bgx1 bgy1 bgz1]-[agx1 agy1 agz1])';

g_min_n(:,i) = [agx1 agy1 agz1]' + m_min_n(i)*...
...([bgx1 bgy1 bgz1]-[agx1 agy1 agz1])';

g_max_n(:,i) = [agx1 agy1 agz1]' + m_max_n(i)*...
...([bgx1 bgy1 bgz1]-[agx1 agy1 agz1])';

g_ges_p = [g_min_p g_max_p];
g_ges_n = [g_min_n g_max_n];

g_ges = [g_ges_p g_ges_n];

%%%Schnittpunkte der Geraden gh mit den Kugelebenen

yr_gerade_min_p(i) = (sqrt(-agz^2 - (agx - a_prime_x(i))^2...
... + rho_min_limit^2) + a_prime_y(i));

yr_gerade_max_p(i) = (sqrt(-agz^2 - (agx - a_prime_x(i))^2...
... + rho_max_limit^2) + a_prime_y(i));

yr_gerade_min_n(i) = (-sqrt(-agz^2 - (agx - a_prime_x(i))^2...
... + rho_min_limit^2) + a_prime_y(i));

yr_gerade_max_n(i) = (-sqrt(-agz^2 - (agx - a_prime_x(i))^2...
... + rho_max_limit^2) + a_prime_y(i));

yr_gerade_min = [yr_gerade_min_p yr_gerade_min_n];
yr_gerade_max = [yr_gerade_max_p yr_gerade_max_n];

% Zusammenfassen in Vektor yr_gerade
yr_gerade = [yr_gerade_min_p yr_gerade_min_n...
... yr_gerade_max_p yr_gerade_max_n];
```



```
end

% Graphische Darstellung
figure
xlabel('y'),ylabel('x'), title('Arbeitsraum');
grid on
hold on

% Darstellung der Kreisebenen in der xy-Ebene
for i = 1:length(yx_ref)

    plot(yx_ref{2,i},yx_ref{1,i})

end

axis([-2 2 -2 2])

%%%%%%%%%%%%Abstandsberechnung

% Abstand Gerade gv
% Ermittlung des kürzesten Intervalls der Schnittpunkte

for j= 1:length(g_ges_p)

    if (g_ges_p(1,j)) == min((g_ges_p(1,:)))

        x_erg_p = g_ges_p(1,j);
        plot(y_dist,g_ges_p(1,j),'*', 'Color',[1 0 0]);

    end

    if (g_ges_n(1,j)) == min((g_ges_n(1,:)))

        x_erg_n = g_ges_n(1,j);
        plot(y_dist,g_ges_n(1,j),'*', 'Color',[1 0 0]);

    end

end
```

```
end

% Referenzpunkt definieren und mit x_erg subtrahieren

x_dist_pos = abs(x_erg_p-(R(1)));
x_dist_neg = abs(x_erg_n-(R(1)));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Abstand Gerade gh
% Ermittlung des kürzesten Intervalls der Schnittpunkte

for i =1:length(yr_gerade)

    if abs(yr_gerade(1,i)) == min(abs(yr_gerade))
        y_erg = yr_gerade(1,i);
        plot(yr_gerade(1,i),agx,'*', 'Color',[1 0 0]);
    end

end

end

% Referenzpunkt-Koordinaten mit y_erg_p und y_erg_n subtrahieren

y_dist_pos = abs(y_erg -(-R(2)));
y_dist_neg = abs(y_erg - R(2));

% Darstellung des Referenzpunktes
plot(y_dist,x_dist,'*', 'Color',[0 1 0])
```

D Berechnung der Abstände des Schwenkwinkels φ im OW

```
clc;
clear all;

% Koordinaten der Referenzposition der Plattform
px = 0;
py = 0;
z = 3.95;

% Minimale und maximale Aktuatorlänge
rho_min_limit = 4.45;
rho_max_limit = 6.45;

% Definition der Orientierung der Plattform in [°]
phi = -25:0.5:25;
theta = 0;
psi = 0;

% Koordinaten der Referenzposition der Plattform
pphi = 0; % [°]

% Umrechnung von [°] in rad/s
phi = phi*pi/180;
theta = theta*pi/180;
psi = psi*pi/180;

% Abstände der Gelenke (a) zum Inertialsystem
% Abstände der Gelenke (b)
% zum körperfesten Koordinatensystem an der Plattform
[a,b] = eingang_Gelenk_K0;

vec_phi = [];

for p = 1:length(phi)
```

```
[rho] = position_winkel_phi...
...([px, py, z, phi(p)], [a], [b]);

% Untersuchung der Punkte für den Arbeitsraum,
% durch Einhalten der technischen Limitierung
if ((min(rho) > rho_min_limit) &&...
... (max(rho) < rho_max_limit));

    vec_phi = [vec_phi;phi(p)];
end
end

% Differenz von maximalen und minimalen Wert
% mit der Referenzposition
v_ref1 = vec_phi(1)*180/pi;
v_ref2 = vec_phi(length(vec_phi))*180/pi;

% Positiver und negativer Abstand zum Arbeitsraumrand
phi_neg = v_ref1-pphi;
phi_pos = v_ref2-pphi;
```